

## WO0157720

Publication Title:

METHODS AND SYSTEMS OF AUTOMATED CLIENT-SERVER DATA  
VALIDATION

Abstract:

Abstract of WO0157720

The present invention provides a way to validate data entry at both the client and the server with minimal engineering effort. The system allows the specification of data validation rules for a form on a server. The system provides a version of the rules with the form to the client computer. The system also executes the rules on the server for further validation. The system may translate the rules into JavaScript data structures, passed as the version to the client for client-side validation. Because the rules for the form are specific only once, there is no possibility of the rules becoming out of sync between the form on the client and on e2f the server. Data supplied from the esp@cenet database - Worldwide

-----  
Courtesy of <http://v3.espacenet.com>

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
9 August 2001 (09.08.2001)

PCT

(10) International Publication Number  
**WO 01/57720 A2**

(51) International Patent Classification<sup>7</sup>: **G06F 17/24**

(21) International Application Number: PCT/US01/03050

(22) International Filing Date: 30 January 2001 (30.01.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:  
60/180,440 4 February 2000 (04.02.2000) US

(71) Applicant (*for all designated States except US*): **AMERICA ONLINE INCORPORATED** [US/US]; 22000 AOL Way, Dulles, VA 20166-9323 (US).

(72) Inventor; and

(75) Inventor/Applicant (*for US only*): **CHOKSHI, Sandip** [US/US]; 4668 Park Arcadia Drive, San Jose, CA 95136 (US).

(74) Agents: **GLENN, Michael** et al.; Glenn Patent Group, 3475 Edison Way, Ste. L., Menlo Park, CA 94025 (US).

(81) Designated States (*national*): AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

**Published:**

— *without international search report and to be republished upon receipt of that report*

*For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

(54) Title: METHODS AND SYSTEMS OF AUTOMATED CLIENT-SERVER DATA VALIDATION

(57) Abstract: The present invention provides a way to validate data entry at both the client and the server with minimal engineering effort. The system allows the specification of data validation rules for a form on a server. The system provides a version of the rules with the form to the client computer. The system also executes the rules on the server for further validation. The system may translate the rules into JavaScript data structures, passed as the version to the client for client-side validation. Because the rules for the form are specific only once, there is no possibility of the rules becoming out of sync between the form on the client and on the server.



**WO 01/57720 A2**

## METHODS AND SYSTEMS OF AUTOMATED CLIENT-SERVER DATA VALIDATION

### Technical field

This invention relates to checking the validity of data entry in web-based  
5 systems.

### Background Art

In web-based systems involving data entry, there is a problem in checking the  
validity of the data. The prior art provides three choices, each having  
disadvantageous consequences:

- 10 1. Check the data only at the client. This lets you be responsive to the  
user (you can display helpful messages in a timely way), but leaves  
your system open to bad data from entities that bypass your client and  
speak directly to your server.
2. Check the data only at the server. This ensures that the server does  
15 not accept bad data, no matter what client is used to send it, but is  
harder to use because it may take a while to display messages to the  
user when data is incorrectly entered.
3. Check the data at the client and at the server. This is responsive to the  
user and is safe from bad data, but requires a lot of engineering work  
20 because the data validation code has to be maintained on two  
platforms.

### Summary of the invention

The present invention provides a way to validate data entry at both the client  
and the server with minimal engineering effort.

- 25 The system allows engineers to specify data validation rules for a form on a  
server. The system provides a version of the rules with the form to the client

computer. The system also executes the rules on the server for further validation. The system may translate the rules into JavaScript data structures, passed as the version to the client for client-side validation. Because engineers only specify the rules for the form once, there is no possibility of the  
5 rules becoming out of sync between the form on the client and on the server.

A set of validation rules can be specified for each form (web page where data entry is allowed). Each rule is associated with a form element (e.g. the users email address or a phone number) and defines what is allowed and what is required for that form element.

10 Each rule can be created from a set of reusable, configurable constraints.

There is a set of pre-written constraints. New constraints can be added. Currently there are number constraints and string constraints.

Number constraints produce rules that force a field to be numeric and may also specify minimum and/or maximum values that the user must meet. String  
15 constraints produce rules that may have minimum or maximum lengths, may allow only certain characters, may require certain characters and may or may not allow white space (e.g. spaces, tabs).

Each supported constraint consists of server-side Java code and client-side JavaScript code.

20 When the server sends the form down to the browser it includes some JavaScript data structures, which are automatically generated from the set of rules that were defined for the form. When the user attempts to submit the form, some JavaScript code first iterates through each rule that was included in the automatically generated data structures. Each rule is checked against  
25 the appropriate entry field. If there is a violation of the rule, the JavaScript code immediately displays a message without involving the server. The generated data structure includes message text for each rule that can be displayed if the user violates a rule.

When the server receives a posted form, it iterates through each of the rules defined for that set. If there is a violation of the rule, the server responds to the client with an error message.

These and other advantages of the present invention will become apparent upon reading the following detailed descriptions and studying the various figures of the drawings.

### Brief Description of the Drawings

Figure 1 depicts a system supporting server computer 4 communicatively coupled 40-42 to client computer 50 operated 52 by client 54, in accordance with certain embodiments of the invention;

Figure 2A depicts a detail flowchart of operation 100 of Figure 1 further performing validating client 54 data entry 52 by a client computer 50 and by a server computer 40 based upon a form 10 containing at least one form entry 12 with at least one associated validation rule 14;

Figure 2B depicts a detail flowchart of operation 132 of Figure 2A further performing the server computer validating the client form data entry received from the client computer;

Figure 3A depicts a detail flowchart of operation 122 of Figure 2A further performing the client computer processing the client form entry;

Figure 3B depicts a detail flowchart of operation 182 of Figure 3A further performing the client computer sending the initial client form data entry to the server computer;

Figure 4A depicts a detail flowchart of operation 216 of Figure 2B further performing the client computer sending the initial client form data entry to the server computer;

Figure **4B** depicts a detail flowchart of operation **152** of Figure **2B** further performing the server computer receiving the client form data entry from the client computer;

5 Figure **5** depicts a server system **2** including server computer **4** of Figure **1** communicating **40-42** with client computer **50** of Figure **1**;

Figure **6** depicts server system **2** of Figure **5** further including server computer **74** communicating **44-42** with client computer **50** of Figure **1** and including server computer **84** with accessibly coupled **86** memory **88** containing form **10**;

10 Figure **7A** depicts a detail flowchart of operation **100** of Figure **2A** further performing validating client data entry by the client computer and by the server computer;

15 Figure **7B** depicts a detail flowchart of operation **112** of Figure **2A** further performing the client computer receiving the one form entry and at least one associated validation rule;

Figure **8A** depicts a detail flowchart of operation **272** of Figure **7A** further performing the second server computer providing the client computer the version of all the associated validation rules of the form entry;

20 Figure **8B** depicts a detail flowchart of operation **112** of Figure **2A** further performing the client computer creating the client form entry with at least one associated local client rule;

Figure **9A** depicts a detail flowchart of operation **272** of Figure **7A** further performing the second server computer providing the client computer the client form entry;

25 Figure **9B** depicts a detail flowchart of operation **122** of Figure **2A** further performing the client computer processing the client form entry;

Figure **10A** depicts a detail flowchart of operation **100** of Figure **2A** further performing the method validating client data entry at both the client computer and the server computer;

5 Figure **10B** depicts a detail flowchart of operation **412** of Figure **10A** further performing maintaining the form collection;

Figure **11A** depicts a flowchart of preparing and sending a first form containing at least one form entry with at least one associated validation rule for use in the methods using forms with both client validation and server validation of form entries with validation rules;

10 Figure **11B** depicts a detail flowchart of operation **452** of Figure **11A** further performing developing the first form and specifying the associated validation rule set for the first form;

Figure **12A** depicts a detail flowchart of operation **112** of Figure **2A** further performing the client computer receiving the client form entry;

15 Figure **12B** depicts a detail flowchart of operation **132** of Figure **2A** further performing the server computer validating the client form data entry;

Figure **13** depicts a detail flowchart of operation **100** of Figure **1** further performing the method of using the form and validating form entries by both server computer **4** and client computer **50**;

20 Figure **14A** depicts a detail flowchart of operation **602** of Figure **13** further performing client computer operations supporting the method of using the form by both server computer **4** and client computer **50**;

Figure **14B** depicts a detail flowchart of operation **612** of Figure **13** further performing server computer operations supporting the method of using the form by both server computer **4** and client computer **50**;

25

Figure **15A** depicts a detail flowchart of operation **632** of Figure **14** further performing the client computer receiving the client form; and

Figure **15B** depicts a detail flowchart of operation **632** of Figure **14** further performing the client computer receiving the client form.

#### Detailed Description of the Invention

5 The present invention provides a way to validate data entry at both the client and the server with minimal engineering effort. The system allows engineers to specify data validation rules for each form or web page where data entry is allowed.

Figure **1** depicts a system supporting server computer **4** communicatively coupled **40-42** to client computer **50** operated **52** by client **54**, in accordance  
10 with certain embodiments of the invention.

Server computer **4** is accessibly coupled **6** to memory **8** containing one form entry **12** associated with at least server rule **14** and at least one client rule **16** based upon form **10**.

Form **10** may further comprise a second form entry **22** associated with at least  
15 one server rule **24** and at least one client rule **26**.

There may be a second form **30** comprising at least one form entry **32** associated with at least server rule **34** and at least one client rule **36**.

Memory **8** may further include program system **100** operating server computer **4** communicating **40-42** with client computer **50** to embody validation of data  
20 entry **52** by client **54** both by the client computer **50** using local client rules **66** and server computer **4** using server rules **44**.

Memory **8** may include, but is not limited to, volatile memory and non-volatile memory. Memory **8** may include, but is not limited to, at least two memory components. The memory components may include volatile memory  
25 components and non-volatile memory components. At least one memory component may be organized as a file management system. The file management system, memory component may include, but is not limited to, at



least one of the following: a semiconductor memory device, a disk drive and a disk array.

Server computer **4** communicatively couples **40-42** to client computer **50**. Where arrows **40** and **42** meet may include, but is not limited to, a gateway, which may or may not possess redundant portals. Where arrows **40** and **42** meet may include, but is not limited to, a firewall. Where arrows **40** and **42** meet may include, but is not limited to, a web site.

As used herein, a computer will include, but is not limited to, instruction processors operating on one or more data streams concurrently and may execute one or more concurrent instruction streams. A computer will include, but is not limited to, inferential engines controlled by an inferential state. The inferential state may be changed by alterations in the active inferential rules, facts contained in a fact database and facts derived from interactions such as client data entry.

As used herein, the operational steps of the methods of this invention may be performed in finite state machines. The operational steps of the methods of this invention may be performed as program steps residing in memory accessibly coupled to a collection of one or more computers. Program steps may be interpreted or natively executed by the computers. Program steps may represent instructions in machine portable interpreted languages including, but not limited to various forms of one or more of the following: HTTP, XML, JAVA, Prolog, LISP, Scheme, C, C++, Smalltalk, FORTRAN and COBOL.

The program steps may implement machine language operations of a computer, interpreted instructions or inferential rules or facts carried out by at least one computer in the systems described herein.

Figure **2A** depicts a detail flowchart of operation **100** of Figure **1** further performing validating client **54** data entry **52** by a client computer **50** and by a

server computer **40** based upon a form **10** containing at least one form entry **12** with at least one associated validation rule **14**.

Arrow **110** directs the flow of execution from starting operation **100** to operation **112**. Operation **112** performs the client computer receiving the form entry and at least one associated validation rule contained in the form to create the client form entry with at least one associated local client rule. Arrow **114** directs execution from operation **112** to operation **116**. Operation **116** terminates the operations of this flowchart.

Arrow **120** directs the flow of execution from starting operation **100** to operation **122**. Operation **122** performs the client computer processing the client form entry using the client data entry and satisfying all of the associated local client rules to create a client form data entry received by the server computer. Arrow **124** directs execution from operation **122** to operation **116**. Operation **116** terminates the operations of this flowchart.

Arrow **130** directs the flow of execution from starting operation **100** to operation **132**. Operation **132** performs the server computer validating the client form data entry received from the client computer based upon all of the associated validation rules of the form entry contained in the form. Arrow **134** directs execution from operation **132** to operation **116**. Operation **116** terminates the operations of this flowchart.

Figure **2B** depicts a detail flowchart of operation **132** of Figure **2A** further performing the server computer validating the client form data entry received from the client computer.

Arrow **150** directs the flow of execution from starting operation **132** to operation **152**. Operation **152** performs the server computer receiving the client form data entry from the client computer. Arrow **154** directs execution from operation **152** to operation **156**. Operation **156** performs the server computer validating the client form data entry based upon all of the associated validation rules of the form entry contained in the form. Arrow **158** directs

execution from operation **156** to operation **160**. Operation **160** terminates the operations of this flowchart.

Note that as used herein, the flowcharts of this document depict an illustrative manner by which to understand the operations of the various methods of certain embodiments of the invention.

In certain embodiments of the invention, concurrent object paradigms, often supporting real-time event-driven operating system may be supported. Such embodiments may further support message passing as a way to trigger events. As such, an arrow and the flow of execution represented by that arrow may serve to depict disparate actions in a variety of distinct objects, which taken in their entirety form an embodiment of the invention.

Note that operation **152** may be performed a different number of times than operation **156**, in such embodiments of the invention utilizing real-time event-driven operating environments for concurrent objects supporting messaging, such as JAVA and C++.

In certain embodiments of the invention supporting inferential engines, these flowcharts portray the inferential action of the components of the inference system embodied.

Figure **3A** depicts a detail flowchart of operation **122** of Figure **2A** further performing the client computer processing the client form entry.

Arrow **170** directs the flow of execution from starting operation **122** to operation **172**. Operation **172** performs the client computer processing the client form entry based upon input from the client to create an initial client form data entry. Arrow **174** directs execution from operation **172** to operation **176**. Operation **176** terminates the operations of this flowchart.

Arrow **180** directs the flow of execution from starting operation **122** to operation **182**. Operation **182** performs the client computer sending the initial client form data entry to the server computer to create the client form data

entry whenever the initial client form data entry satisfies all of the associated local client rules. Arrow **184** directs execution from operation **182** to operation **176**. Operation **176** terminates the operations of this flowchart.

Figure **3B** depicts a detail flowchart of operation **182** of Figure **3A** further performing the client computer sending the initial client form data entry to the server computer.

Arrow **210** directs the flow of execution from starting operation **182** to operation **212**. Operation **212** determines when the initial client form data entry satisfies all of the associated local client rules. Arrow **214** directs execution from operation **212** to operation **216** when the determination is 'Yes'. Arrow **228** directs execution to **220** when the determination is 'No'.

Operation **216** performs the client computer sending the initial client form data entry to the server computer to create the client form data entry. Arrow **218** directs execution from operation **216** to operation **220**. Operation **220** terminates the operations of this flowchart.

Figure **4A** depicts a detail flowchart of operation **216** of Figure **2B** further performing the client computer sending the initial client form data entry to the server computer.

Arrow **230** directs the flow of execution from starting operation **216** to operation **232**. Operation **232** performs the client encrypting the initial client form data entry to create an encrypted data entry. Arrow **234** directs execution from operation **232** to operation **236**. Operation **236** performs the client sending the encrypted data entry to the server computer. Arrow **238** directs execution from operation **236** to operation **240**. Operation **240** terminates the operations of this flowchart.

Figure **4B** depicts a detail flowchart of operation **152** of Figure **2B** further performing the server computer receiving the client form data entry from the client computer.

Arrow **250** directs the flow of execution from starting operation **152** to operation **252**. Operation **252** performs the server receiving the encrypted data entry. Arrow **254** directs execution from operation **252** to operation **256**. Operation **256** performs the server computer decrypting the encrypted data entry to create the client form data entry. Arrow **258** directs execution from operation **256** to operation **260**. Operation **260** terminates the operations of this flowchart.

Figure **5** depicts a server system **2** including server computer **4** of Figure **1** communicating **40-42** with client computer **50** of Figure **1**.

Memory **8** may contain form **10**.

Memory **58** may include a web browser **60**. Web browser **60** may further perform client operations supporting the program system **100**. Note that in certain embodiments of the invention, the browser **60** may download a version of those client operations and form as a web page.

Memory **58** may include client form **10**.

Note that rule **16** is associated with form entry **12** and form entry **22**. An example of a rule which is associated with more than one form entry would be a rule which validates that birth date and age must be close to each other. A birth date entry indicating an age of 16 and an age entry of 21 should fail such a validation rule.

Figure **6** depicts server system **2** of Figure **5** further including server computer **74** communicating **44-42** with client computer **50** of Figure **1** and including server computer **84** with accessibly coupled **86** memory **88** containing form **10**.

Server computer **74** may provide client computer **50** with client form entry **62** and associated local client rules **66**.

Server computer **84** may provide server rules **14** to server computer **4**. Server computer **84** may further provide form **10** to server computer **74**.

Server computer **84** may provide a version of form **10** to server computer **74**.  
Server computer **84** may generate a version of form **10**.

Server computer **74** may generate a version of form **10** to send to client computer **50**.

- 5     Note that in certain embodiments, form **10** may include form entries **12** with pre-computed versions of the server rules **14** and client rules **16**.

Figure **7A** depicts a detail flowchart of operation **100** of Figure **2A** further performing validating client data entry by the client computer and by the server computer.

- 10     Arrow **270** directs the flow of execution from starting operation **100** to operation **272**. Operation **272** performs a second server computer providing the client computer a version of the form entry and at least one associated validation rule. Arrow **274** directs execution from operation **272** to operation **276**. Operation **276** terminates the operations of this flowchart.

- 15     Figure **7B** depicts a detail flowchart of operation **112** of Figure **2A** further performing the client computer receiving the one form entry and at least one associated validation rule.

- Arrow **290** directs the flow of execution from starting operation **112** to operation **292**. Operation **292** performs the client computer receiving the one  
20     form entry and all of the associated validation rules from the second server computer to create the client form entry with at least one associated local client rule. Arrow **294** directs execution from operation **292** to operation **296**. Operation **296** terminates the operations of this flowchart.

- Figure **8A** depicts a detail flowchart of operation **272** of Figure **7A** further  
25     performing the second server computer providing the client computer the version of all the associated validation rules of the form entry.

Arrow **310** directs the flow of execution from starting operation **272** to operation **312**. Operation **312** performs the second server computer

generating a version of all the associated validation rules based upon the one form entry and all of the associated validation rules. Arrow **314** directs execution from operation **312** to operation **316**. Operation **316** performs the second server computer sending the version to the client computer to create  
5 all the associated local client rules. Arrow **318** directs execution from operation **316** to operation **320**. Operation **320** terminates the operations of this flowchart.

Figure **8B** depicts a detail flowchart of operation **112** of Figure **2A** further performing the client computer creating the client form entry with at least one  
10 associated local client rule.

Arrow **330** directs the flow of execution from starting operation **112** to operation **332**. Operation **332** performs the client computer receiving the one form entry to create the client form entry. Arrow **334** directs execution from operation **332** to operation **336**. Operation **336** performs the client computer  
15 receiving the version of all the associated validation rules to create all the local client rules associated with the client form entry. Arrow **338** directs execution from operation **336** to operation **340**. Operation **340** terminates the operations of this flowchart.

Note that in certain embodiments of the invention, a version of the form entry  
20 may be received as well.

Figure **9A** depicts a detail flowchart of operation **272** of Figure **7A** further performing the second server computer providing the client computer the client form entry.

Arrow **350** directs the flow of execution from starting operation **272** to  
25 operation **352**. Operation **352** performs the second server computer providing the client computer the form entry and all the associated validation rules to create the client form entry with all the associated local client rules, for each of the form entries with the associated validation rules contained in the form.

Arrow **354** directs execution from operation **352** to operation **356**. Operation **356** terminates the operations of this flowchart.

Figure **9B** depicts a detail flowchart of operation **122** of Figure **2A** further performing the client computer processing the client form entry.

5 Arrow **370** directs the flow of execution from starting operation **122** to operation **372**. Operation **372** performs the client computer receiving a submission request from the client to create a form submission request. Arrow **374** directs execution from operation **372** to operation **376**. Operation **376** terminates the operations of this flowchart.

10 Arrow **380** directs the flow of execution from starting operation **122** to operation **382**. Operation **382** performs the client computer validating the client form entry based upon all of the associated local client rules to create a validated client form data entry, whenever the client computer receives the form submission request, for each form entry with at least one associated  
15 validation rule contained in the form. Arrow **384** directs execution from operation **382** to operation **376**. Operation **376** terminates the operations of this flowchart.

Arrow **390** directs the flow of execution from starting operation **122** to operation **392**. Operation **392** performs the client computer sending all of the  
20 validated client form data entries to the server computer to create all of the client form data entries for the form whenever the initial client form data entries satisfy all of the associated local client rules, for all of the at least one form entry with at least one associated validation rule contained in the form. Arrow **394** directs execution from operation **392** to operation **376**. Operation  
25 **376** terminates the operations of this flowchart.

Figure **10A** depicts a detail flowchart of operation **100** of Figure **2A** further performing the method validating client data entry at both the client computer and the server computer.



Arrow **410** directs the flow of execution from starting operation **100** to operation **412**. Operation **412** performs maintaining a form collection including at least one form comprised of at least one form entry with at least one associated validation rule. Arrow **414** directs execution from operation **412** to operation **416**. Operation **416** terminates the operations of this flowchart.

Figure **10B** depicts a detail flowchart of operation **412** of Figure **10A** further performing maintaining the form collection.

Arrow **430** directs the flow of execution from starting operation **412** to operation **432**. Operation **432** performs receiving a first form and specifying an associated validation rule set for the first form from a development computer. Arrow **434** directs execution from operation **432** to operation **436**. Operation **436** performs adding the first form and the associated validation rule set into the form collection. Arrow **438** directs execution from operation **436** to operation **440**. Operation **440** terminates the operations of this flowchart.

Certain embodiments of the invention may include preparing a form containing at least one form entry with at least one associated validation, and sending that form to be added to a form collection.

Such operations may be performed as program steps residing in memory accessibly coupled to a development computer. The development computer may further be a server development computer or a client development computer. The server development computer may further belong to the server system containing the server computer. The server development computer may further be any of the following: the server computer **4** the second server computer **74** or the third server computer **84**.

Sending the form to be added to the maintained form collection may further include sending the form as a message to a method of an object of a class

maintaining a form collection available to server computer **4**. The form collection may further be available to second server computer **74**.

Figure **11A** depicts a flowchart of preparing and sending a first form containing at least one form entry with at least one associated validation rule for use in the methods using forms with both client validation and server validation of form entries with validation rules.

Arrow **450** directs the flow of execution from starting operation **100** to operation **452**. Operation **452** performs developing the first form and specifying an associated validation rule set for the first form. Arrow **454** directs execution from operation **452** to operation **456**. Operation **456** terminates the operations of this flowchart.

Arrow **460** directs the flow of execution from starting operation **100** to operation **462**. Operation **462** performs sending the first form and the associated validation rule set for addition into the form collection. Arrow **464** directs execution from operation **462** to operation **456**. Operation **456** terminates the operations of this flowchart.

Note that the associated validation rule set may be comprised of at least one associated validation rule.

Each of the associated validation rules may be associated with at least one form entry of the form collection.

Each rule may be created from a set of reusable, configurable constraints.

There may be a set of pre-written constraints. Currently there are number constraints and string constraints.

New constraints may be added.

Number constraints produce rules that force a field to be numeric and may also specify minimum and/or maximum values that the user must meet. String constraints produce rules that may have minimum or maximum lengths, may

allow only certain characters, may require certain characters and may or may not allow white space (e.g. spaces, tabs).

Each supported constraint consists of server-side Java code and client-side JavaScript code.

- 5 The system may translate the rules of the form into JavaScript data structures, which are passed to the client for client-side validation of the form entries and the system also executes the rules on the server for further validation of the form entries. Because engineers only specify the rules once for the form, there is no possibility of the rules becoming out of sync between  
10 the form on the client and on the server.

Figure **11B** depicts a detail flowchart of operation **452** of Figure **11A** further performing developing the first form and specifying the associated validation rule set for the first form.

- Arrow **470** directs the flow of execution from starting operation **452** to  
15 operation **472**. Operation **472** performs maintaining a form entry collection containing at least one form entry. Arrow **474** directs execution from operation **472** to operation **476**. Operation **476** terminates the operations of this flowchart.

- Arrow **480** directs the flow of execution from starting operation **452** to  
20 operation **482**. Operation **482** performs specifying the associated validation rule set based upon the form entry collection. Arrow **484** directs execution from operation **482** to operation **476**. Operation **476** terminates the operations of this flowchart.

- Figure **12A** depicts a detail flowchart of operation **112** of Figure **2A** further  
25 performing the client computer receiving the client form entry.

Arrow **490** directs the flow of execution from starting operation **112** to operation **492**. Operation **492** performs the client computer receiving the step **122** of the client computer processing the client form entry. Arrow **494** directs

execution from operation **492** to operation **496**. Operation **496** terminates the operations of this flowchart.

The server may send the form to the client computer. The client computer may be operating a browser receiving the sent form as shown in Figure 5. The form may include some JavaScript data structures. The JavaScript structures may be automatically generated from the set of rules defined for the form.

When the user attempts to submit the form, some JavaScript code may first iterate through each rule that was included in the automatically generated data structures.

Each rule is checked against the appropriate entry field. If there is a violation of the rule, the JavaScript code immediately displays a message without involving the server. The generated data structure includes message text for each rule that can be displayed if the user violates a rule.

Figure **12B** depicts a detail flowchart of operation **132** of Figure **2A** further performing the server computer validating the client form data entry.

Arrow **510** directs the flow of execution from starting operation **132** to operation **512**. Operation **512** performs the server computer receiving all the associated server validation rules based upon the form entry with at least one associated validation rule. Arrow **514** directs execution from operation **512** to operation **516**. Operation **516** terminates the operations of this flowchart.

Arrow **520** directs the flow of execution from starting operation **132** to operation **522**. Operation **522** performs the server computer validating the client form data entry based upon all the associated server validation rules. Arrow **524** directs execution from operation **522** to operation **516**. Operation **516** terminates the operations of this flowchart.

When the server receives a posted form, it iterates through each of the rules defined for that set. If there is a violation of the rule, the server responds to the client with an error message.

Certain embodiments of the invention include a method of using a form containing at least one form entry with at least one associated validation rule performing data validation by both a server computer and a client computer.

Figure 13 depicts a detail flowchart of operation 100 of Figure 1 further performing the method of using the form and validating form entries by both server computer 4 and client computer 50.

Arrow 600 directs the flow of execution from starting operation 100 to operation 602. Operation 602 performs client computer operations supporting the method using the form and validating the form entries with at least one validation rule. Arrow 604 directs execution from operation 602 to operation 606. Operation 606 terminates the operations of this flowchart.

Arrow 610 directs the flow of execution from starting operation 100 to operation 612. Operation 612 performs server computer operations supporting the method using the form and validating the form entries with at least one validation rule. Arrow 614 directs execution from operation 612 to operation 606. Operation 606 terminates the operations of this flowchart.

Figure 14A depicts a detail flowchart of operation 602 of Figure 13 further performing client computer operations supporting the method using the form and validating the form entries with at least one validation rule.

Arrow 630 directs the flow of execution from starting operation 602 to operation 632. Operation 632 performs receiving a client form containing at least one client form entry with at least one associated local client rule based upon the form containing the at least one form entry with at least one associated validation rule. Arrow 634 directs execution from operation 632 to operation 636. Operation 636 terminates the operations of this flowchart.

Arrow 640 directs the flow of execution from starting operation 602 to operation 642. Operation 642 performs processing client data entry based upon the client form to create a client form data entry for the at least one client

form entry. Arrow **644** directs execution from operation **642** to operation **636**. Operation **636** terminates the operations of this flowchart.

Arrow **650** directs the flow of execution from starting operation **602** to operation **652**. Operation **652** performs receiving a submission request from the client to create a form submission request. Arrow **654** directs execution from operation **652** to operation **636**. Operation **636** terminates the operations of this flowchart.

Arrow **660** directs the flow of execution from starting operation **602** to operation **662**. Operation **662** performs validating the client form entry based upon all the associated local client rules to create a validated client form data entry whenever the client computer receives the form submission request, for each form entry with at least one associated validation rule contained in the form. Arrow **664** directs execution from operation **662** to operation **636**. Operation **636** terminates the operations of this flowchart.

Arrow **670** directs the flow of execution from starting operation **602** to operation **672**. Operation **672** performs sending all of the validated client form data entries to the server computer to create the sent client form data entry for each form entry with at least one associated validation rule, whenever the client requests form submission and whenever all client form data entries satisfy all of the associated local client rules. Arrow **674** directs execution from operation **672** to operation **636**. Operation **636** terminates the operations of this flowchart.

Figure **14B** depicts a detail flowchart of operation **612** of Figure **13** further performing server computer operations supporting the method using the form and validating the form entries with at least one validation rule.

Arrow **770** directs the flow of execution from starting operation **612** to operation **772**. Operation **772** performs receiving the sent client form data entry for each form entry with at least one validation rule. Arrow **774** directs

execution from operation **772** to operation **776**. Operation **776** terminates the operations of this flowchart.

Arrow **780** directs the flow of execution from starting operation **612** to operation **782**. Operation **782** performs validating the sent client form data entries based upon all of the associated validation rules, for all form entries with at least one associated validation rule contained in the form. Arrow **784** directs execution from operation **782** to operation **776**. Operation **776** terminates the operations of this flowchart.

Figure **15A** depicts a detail flowchart of operation **632** of Figure **14** further performing the client computer receiving the client form.

Arrow **710** directs the flow of execution from starting operation **632** to operation **712**. Operation **712** performs the client computer operating a browser downloading a web page based upon the form to create the client form containing a client form entry with all the associated local client rules, for each form entry with at least one associated validation rule contained in the form. Arrow **714** directs execution from operation **712** to operation **716**. Operation **716** terminates the operations of this flowchart.

Figure **15B** depicts a detail flowchart of operation **632** of Figure **14** further performing the client computer receiving the client form.

Arrow **730** directs the flow of execution from starting operation **632** to operation **732**. Operation **732** performs the client computer receiving the step **642** of the client computer processing client data entry based upon the client form. Arrow **734** directs execution from operation **732** to operation **736**. Operation **736** terminates the operations of this flowchart.

Arrow **740** directs the flow of execution from starting operation **632** to operation **742**. Operation **742** performs the client computer receiving the step **652** of the client computer receiving the submission request from the client. Arrow **744** directs execution from operation **742** to operation **736**. Operation **736** terminates the operations of this flowchart.

Arrow **750** directs the flow of execution from starting operation **632** to operation **752**. Operation **752** performs the client computer receiving the step **662** of the client computer validating the client form entry. Arrow **754** directs execution from operation **752** to operation **736**. Operation **736** terminates the operations of this flowchart.

Arrow **760** directs the flow of execution from starting operation **632** to operation **762**. Operation **762** performs the client computer receiving the step **672** of the client computer sending all of the validated client form data entries. Arrow **764** directs execution from operation **762** to operation **736**. Operation **736** terminates the operations of this flowchart.

The preceding embodiments have been provided by way of example and are not meant to constrain the scope of the following claims.



### Claims

1. A method of using a form containing at least one form entry with at least one associated validation rule performing data validation by both a server computer and a client computer comprising the steps of:

5 the client computer performing the steps of:

receiving a client form containing at least one client form entry with all associated local client rules based upon the form containing the at least one form entry with the at least one associated validation rule;

10 processing client data entry based upon the client form to create a client form data entry for the client form entry;

receiving a submission request from the client to create a form submission request;

15 validating the client form entry based upon all the associated local client rules to create a validated client form data entry whenever the client computer receives the form submission request, for each form entry with at least one associated validation rule contained in the form;

20 sending all of the validated client form data entries to the server computer to create the sent client form data entry for each form entry with at least one associated validation rule, whenever the client requests form submission and whenever all client form data entries satisfy all of the associated local client rules;

the server computer performing the steps of:

25 receiving the sent client form data entry for each form entry with at least one validation rule;

validating the sent client form data entries based upon all of the associated validation rules, for all form entries with at least one associated validation rule contained in the form.

30 2. The method of Claim 1,

wherein the step of the client computer receiving the client form is further comprised of the step of:

the client computer operating a browser downloading a web page based upon the form to create the client form containing a client form entry with all the associated local client rules, for each form entry with at least one associated validation rule contained in the form.

5

3. The method of Claim 1,

wherein the step of the client computer receiving the client form further comprised of the steps of:

the client computer receiving the step of the client computer processing  
10 client data entry based upon the client form;

the client computer receiving the step of the client computer receiving the submission request from the client;

the client computer receiving the step of the client computer validating the client form entry; and

15 the client computer receiving the step of the client computer sending all of the validated client form data entries.

4. A program system controlling a server computer communicating with a client computer to validate client data entry by both the client computer and by  
20 the server computer by performing the steps of Claim 1 as program steps residing in memory accessibly coupled to at least one member of the collection comprising the server computer and the client computer.

5. A system for using a form containing at least one form entry with an associated validation rule to perform data validation by both a server  
25 computer and by a client computer, comprising:

the server computer communicatively coupled to a client computer; and  
a program system comprising the program steps residing in memory accessibly coupled to at least one member of the collection comprising the  
30 server computer and the client computer;

wherein the program system is comprised of program steps residing in memory accessibly coupled to the client computer of:

receiving a client form containing at least one client form entry with all associated local client rules based upon the form containing the at least one form entry with the at least one associated validation rule;

processing client data entry based upon the client form to create a client form data entry for the client form entry;

receiving a submission request from the client to create a form submission request;

validating the client form entry based upon all the associated local client rules to create a validated client form data entry whenever the client computer receives the form submission request, for each form entry with at least one associated validation rule contained in the form; and

sending all of the validated client form data entries to the server computer to create the sent client form data entry for each form entry with at least one associated validation rule, whenever the client requests form submission and whenever all client form data entries satisfy all of the associated local client rules; and

wherein the program system is comprised of program steps residing in memory accessibly coupled to the server computer of:

receiving the sent client form data entry for each form entry with at least one validation rule; and

validating the sent client form data entries based upon all of the associated validation rules, for all form entries with at least one associated validation rule contained in the form.

6. The system of Claim 5,

wherein the program step of the client computer receiving the client form is further comprised of the program step of:

the client computer operating a browser downloading a web page based upon the form to create the client form containing a client form entry with all the associated local client rules, for each form entry with at least one associated validation rule contained in the form.

7. The system of Claim 5,  
wherein the program step of the client computer receiving the client  
form further comprised of the program steps of:

the client computer receiving the program step of the client computer  
5 processing client data entry based upon the client form;

the client computer receiving the program step of the client computer  
receiving the submission request from the client;

the client computer receiving the program step of the client computer  
validating the client form entry; and

10 the client computer receiving the program step of the client computer  
sending all of the validated client form data entries.

8. A method of validating client data entry by a client computer and by a  
server computer based upon a form containing at least one form entry with at  
15 least one associated validation rule, comprising the steps of:

the client computer receiving the form entry and all of the associated  
validation rules contained in the form to create the client form entry with at  
least one associated local client rule;

the client computer processing the client form entry using the client  
20 data entry and satisfying all of the associated local client rules to create a  
client form data entry received by the server computer; and

the server computer validating the client form data entry received from  
the client computer based upon all of the associated validation rules of the  
form entry contained in the form.

25 9. The method of Claim 8,  
wherein the step of the server computer validating the client form data  
entry received from the client computer is further comprised of the steps of:

the server computer receiving the client form data entry from the client  
30 computer; and

the server computer validating the client form data entry based upon all  
of the associated validation rules of the form entry contained in the form.

10. The method of Claim 9,  
wherein the step of the client computer processing the client form entry  
is further comprised of the steps of:

5 the client computer processing the client form entry based upon input  
from the client to create an initial client form data entry; and

the client computer sending the initial client form data entry to the  
server computer to create the client form data entry whenever the initial client  
form data entry satisfies all of the associated local client rules.

10

11. The method of Claim 10,  
wherein the step of the client computer sending the initial client form  
data entry to the server computer is further comprised of the steps of:

the client encrypting the initial client form data entry to create an  
15 encrypted data entry; and

the client sending the encrypted data entry to the server computer; and

wherein the step of the server computer receiving the client form data  
entry from the client computer is further comprised of the steps of:

the server receiving the encrypted data entry; and

20 the server computer decrypting the encrypted data entry to create the  
client form data entry.

12. The method of Claim 8, further comprising of the step of  
a second server computer providing the client computer a version of  
25 the form entry and all of the associated validation rules; and

wherein the step of the client computer receiving the one form entry  
and all of the associated validation rules is further comprised of the step of:

the client computer receiving the version of the one form entry and all  
of the associated validation rules from the second server computer to create  
30 the client form entry with at least one associated local client rule.

13. The method of Claim 12,

wherein the step of the second server computer providing the client computer the version is further comprised of the steps of:

the second server computer generating a version of all of the associated validation rules based upon the one form entry and all of the associated validation rules; and

the second server computer sending the version to the client computer to create all of the associated local client rules.

14. The method of Claim 12,

wherein the step of the second server computer providing the client computer the client form entry is further comprised of the steps of:

the second server computer providing the client computer the form entry and all the associated validation rules to create the client form entry with all the associated local client rules, for each of the form entries with at least one associated validation rules contained in the form.

15. The method of Claim 14,

wherein the step of the client computer processing the client form entry is further comprised of the steps of:

the client computer receiving a submission request from the client to create a form submission request;

the client computer validating the client form entry based upon all of the associated local client rules to create a validated client form data entry, whenever the client computer receives the form submission request, for each form entry with at least one associated validation rule contained in the form; and

the client computer sending all of the validated client form data entries to the server computer to create all of the client form data entries for the form whenever the initial client form data entries satisfy all of the associated local client rules, for all of the at least one form entry with at least one associated validation rule contained in the form.

16. The method of Claim 12,  
wherein a server system further contains the server computer and the  
second server computer.

5 17. The method of Claim 12, further comprising the step of:  
maintaining a form collection including at least one form comprised of  
at least one form entry with at least one associated validation rule.

18. The method of Claim 17,  
10 wherein the step of maintaining the form collection is further comprised  
of the steps of:

receiving a first form and specifying an associated validation rule set  
for the first form from a development computer; and

15 adding the first form and the associated validation rule set into the form  
collection;

wherein the method further comprises the steps of:

developing the first form and specifying an associated validation rule  
set for the first form; and

20 sending the first form and the associated validation rule set for addition  
into the form collection;

wherein the associated validation rule set is comprised of at least one  
associated validation rule; and

wherein each of the associated validation rules is associated with at  
least one form entry of the form collection.

25

19. The method of Claim 18,  
wherein the step of developing the first form and specifying the  
associated validation rule set for the first form is further comprised of the  
steps of:

30 maintaining a form entry collection containing at least one form entry;  
and

specifying the associated validation rule set based upon the form entry collection.

20. The method of Claim 8,

5 wherein the step of the client computer receiving the client form entry with all the associated local client rules further comprised of the steps of:

the client computer receiving the one form entry to create the client form entry; and

10 the client computer receiving all of the associated validation rules to create all of the at least one local client rules associated with the client form entry.

21. The method of Claim 8,

15 wherein the step of the client computer receiving the client form entry further comprised of the step of:

the client computer receiving the step of the client computer processing the client form entry.

22. The method of Claim 8,

20 wherein the step of the server computer validating the client form data entry is further comprised of the step of:

the server computer receiving all of the associated server validation rules based upon the form entry with the at least one associated validation rule; and

25 the server computer validating the client form data entry based upon all of the associated server validation rules.

!!!

30 23. A program system controlling a server computer communicating with a client computer for validating client data entry by the client computer and by the server computer performing the steps of Claim 8 as program steps



residing in memory accessibly to at least one member of the collection comprising the server computer and the client computer.

24. A system validating of client data entry a client computer and by a server computer based upon a form containing at least one form entry with at least one associated validation rule, comprising:

the server computer communicating with the client computer; and  
a program system controlling the system is comprised of program steps in accessibly coupled memory of at least the members of the collection comprising the server computer and the client computer;

wherein the program system further comprises the program steps residing in accessibly coupled memory to the client computer of:

the client computer receiving the form entry and all of the associated validation rules contained in the form to create the client form entry with at least one associated local client rule;

the client computer processing the client form entry using the client data entry and satisfying all of the associated local client rules to create a client form data entry received by the server computer; and

wherein the program system further comprises the program step residing in accessibly coupled memory to the server computer of:

the server computer validating the client form data entry received from the client computer based upon all of the associated validation rules of the form entry contained in the form.

25. The system of Claim 24,

wherein the program step of the server computer validating the client form data entry received from the client computer is further comprised of the program steps of:

the server computer receiving the client form data entry from the client computer; and

the server computer validating the client form data entry based upon all of the associated validation rules of the form entry contained in the form.

26. The system of Claim 25,

wherein the program step of the client computer processing the client form entry is further comprised of the program steps of:

5 the client computer processing the client form entry based upon input from the client to create an initial client form data entry; and

the client computer sending the initial client form data entry to the server computer to create the client form data entry whenever the initial client form data entry satisfies all of the associated local client rules.

10

27. The system of Claim 26,

wherein the program step of the client computer sending the initial client form data entry to the server computer is further comprised of the program steps of:

15 the client encrypting the initial client form data entry to create an encrypted data entry; and

the client sending the encrypted data entry to the server computer; and

wherein the program step of the server computer receiving the client form data entry from the client computer is further comprised of the program steps of:

20

the server receiving the encrypted data entry; and

the server computer decrypting the encrypted data entry to create the client form data entry.

25 28. The system of Claim 24, further comprising:

a second server computer providing the client computer with a version of the form entry and all of the associated validation rules;

wherein the program system further comprises a program step residing in a memory accessibly coupled to the second server computer of:

30

a second server computer providing the client computer with a version of the form entry and all of the associated validation rules;

wherein the program step of the client computer receiving the one form entry and all of the associated validation rules is further comprised of the program step of:

5 the client computer receiving the version of the one form entry and all of the associated validation rules from the second server computer to create the client form entry with at least one associated local client rule.

29. The system of Claim 28,

10 wherein the program step of the second server computer providing the associated validation rule to the client is further comprised of the program steps of:

the second server computer generating a version of all of the associated validation rules based upon the one form entry and all of the associated validation rules; and

15 the second server computer sending the version to the client computer to create all of the associated local client rules.

30. The system of Claim 28,

20 wherein the program step of the second server computer providing the client computer the client form entry is further comprised of the program steps of:

the second server computer providing the client computer the form entry and all the associated validation rules to create the client form entry with all the associated local client rules, for each of the form entries with at least  
25 one associated validation rules contained in the form.

31. The system of Claim 30,

wherein the program step of the client computer processing the client form entry is further comprised of the program steps of:

30 the client computer receiving a submission request from the client to create a form submission request;

the client computer validating the client form entry based upon all of the associated local client rules to create a validated client form data entry, whenever the client computer receives the form submission request, for each form entry with at least one associated validation rule contained in the form;

5 and

the client computer sending all of the validated client form data entries to the server computer to create all of the client form data entries for the form whenever the initial client form data entries satisfy all of the associated local client rules, for all of the at least one form entry with at least one associated validation rule contained in the form.

10

32. The system of Claim 28, further comprising:

a server system comprising the server computer and the second server computer.

15

33. The system of Claim 28,

wherein the program system further comprises the program step, residing in memory accessibly coupled to at least one server computer contained in the server system, of:

20 maintaining a form collection including at least one form comprised of at least one form entry with at least one associated validation rule.

34. The system of Claim 33, further comprising:

25 a third server computer communicatively coupled with at least one server computer contained in the server system;

wherein the program step of maintaining the form collection is further comprised of the program step residing in memory accessibly coupled to the third server computer.

30 35. The system of Claim 34,

wherein the third server computer is contained in the server system.

36. The system of Claim 24,  
wherein the program step of the client computer processing the client  
form entry is further comprised of the program steps of:

the client computer processing the client form entry based upon input  
5 from the client to create an initial client form data entry; and

the client computer sending the initial client form data entry to the  
server system to create the client form data entry whenever the initial client  
form data entry satisfies all of the associated local client rules.

10 37. The system of Claim 24,  
wherein the program system further comprises the program step of:  
maintaining a form collection including at least one form comprised of  
at least one form entry with at least one associated validation rule.

15 38. The system of Claim 33,  
wherein the program step of maintaining the form collection resides in  
a memory accessibly coupled to a third server computer.

39. The system of Claim 33,  
20 wherein the program step of maintaining the form collection resides in  
a memory accessibly coupled to the server computer.

40. The system of Claim 33,  
wherein the program step of maintaining the form collection is further  
25 comprised of the program steps of:

receiving a first form and specifying an associated validation rule set  
for the first form from a development computer; and

adding the first form and the associated validation rule set into the form  
collection; and

30 wherein the program system is further comprised of the program steps  
residing in memory accessibly coupled to the development computer of:

developing the first form and specifying an associated validation rule set for the first form; and

sending the first form and the associated validation rule set for addition into the form collection;

5        wherein the associated validation rule set is comprised of at least one associated validation rule; and

       wherein each of the associated validation rules is associated with at least one form entry of the form collection.

10    41.    The system of Claim 40,

       wherein the program step of developing the first form and specifying the associated validation rule set is further comprised of the program steps of:

       maintaining a form entry collection containing at least one form entry;

       and

15        specifying the associated validation rule set based upon the form entry collection.

       42.    The system of Claim 24,

       wherein the program step of the client computer creating the client form entry with the at least one associated local client rule further comprised of the program steps of:

       the client computer receiving the one form entry to create the client form entry; and

25        the client computer receiving all of the associated validation rules to create all of the at least one local client rules associated with the client form entry.

       43.    The system of Claim 24,

30        wherein the program step of the client computer receiving the client form entry is further comprised of the program step of:

       the client computer receiving the program step of the client computer processing the client form entry.

44. The system of Claim 24,  
wherein the program step of the server computer validating the client  
form data entry is further comprised of the program step of:

5 the server computer receiving all of the associated server validation  
rules based upon the form entry with the at least one associated validation  
rule; and

the server computer validating the client form data entry based upon all  
of the associated server validation rules.

10

45. The system of Claim 44,  
wherein the program step of the second server computer providing the  
server associated validation rule is further comprised of the program step of:

the second server computer generating all of the server associated  
15 validation rules based upon all of the associated validation rules.

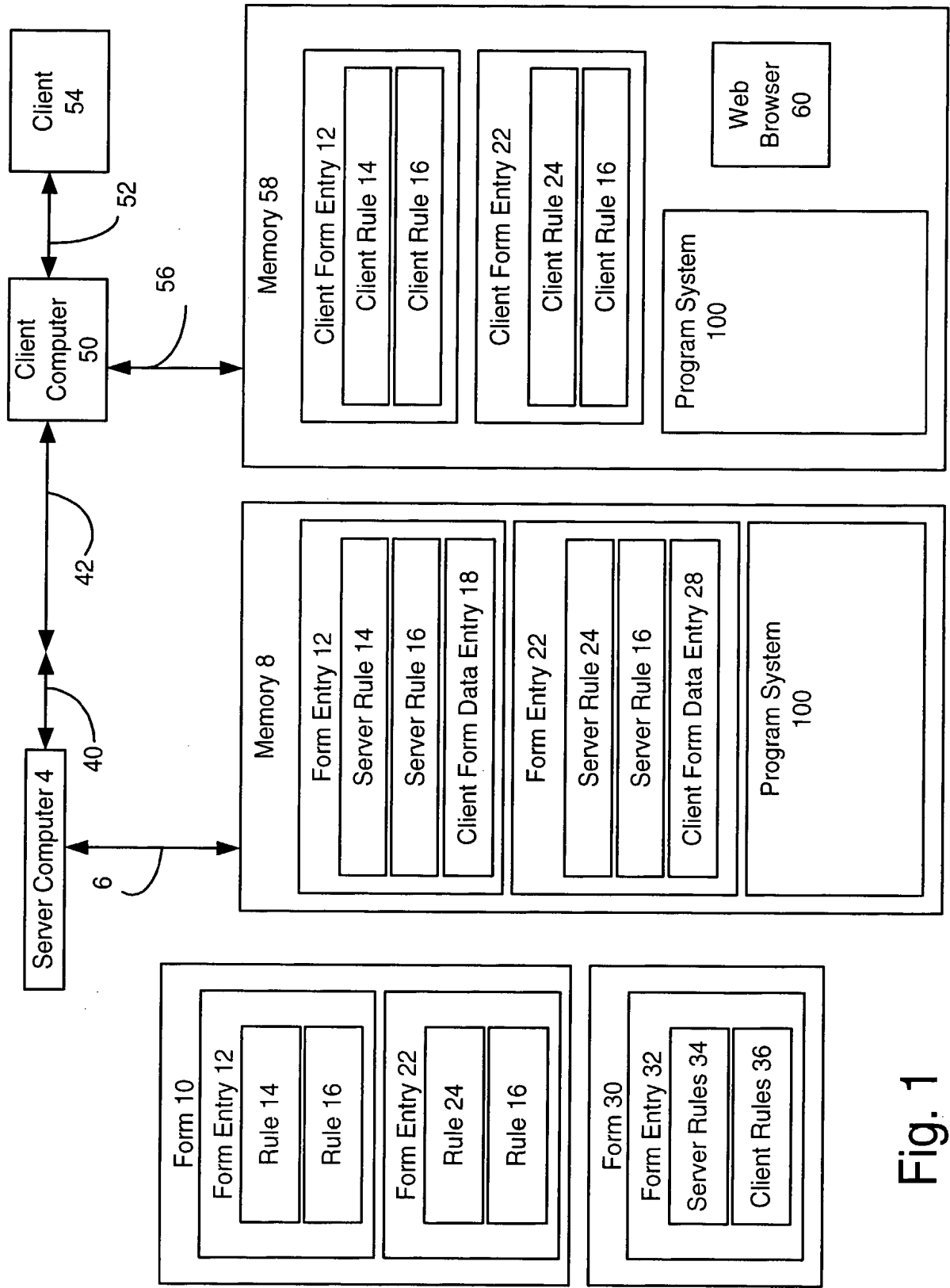


Fig. 1



2/16

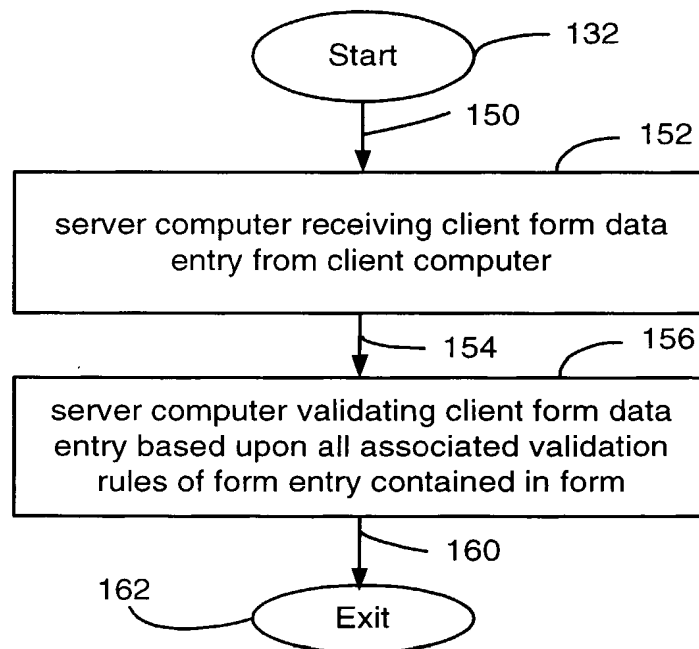
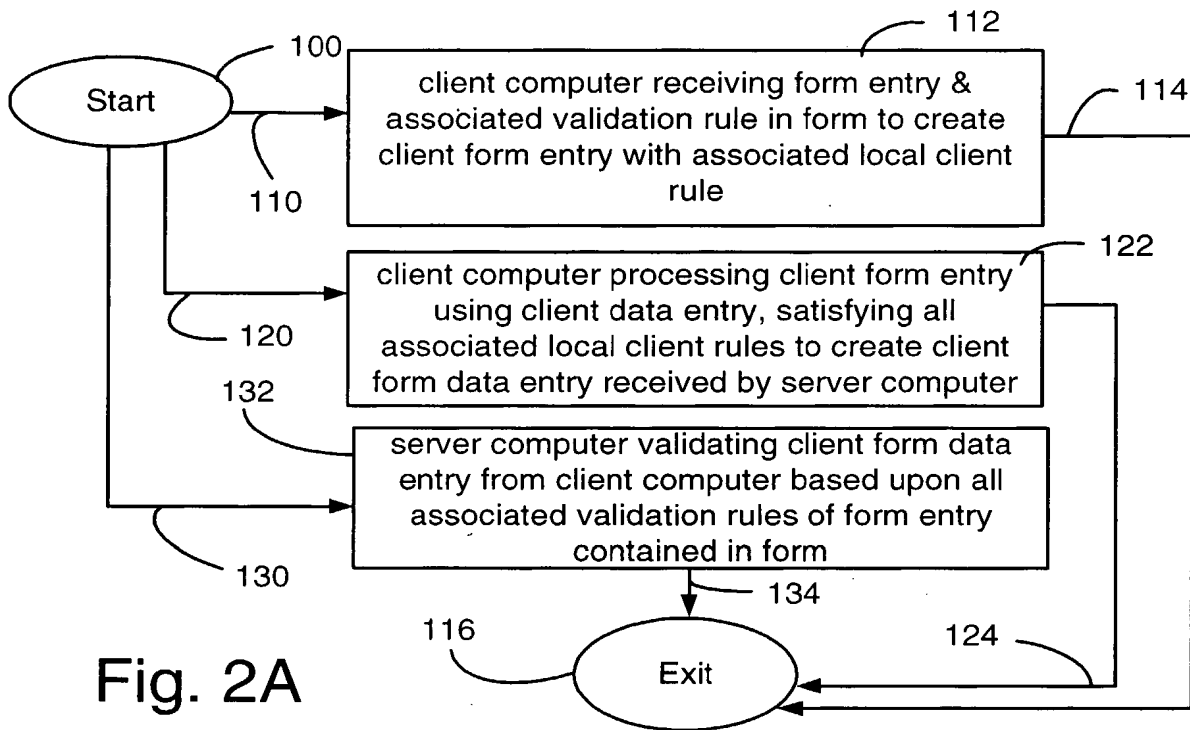


Fig. 2B

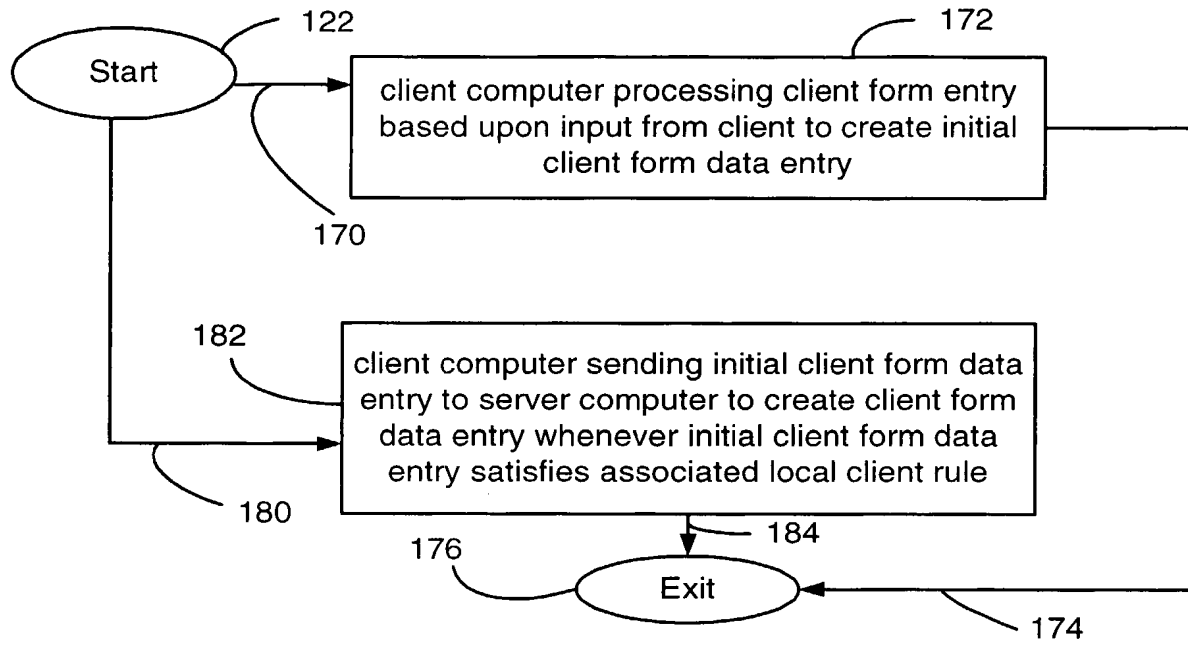


Fig. 3A

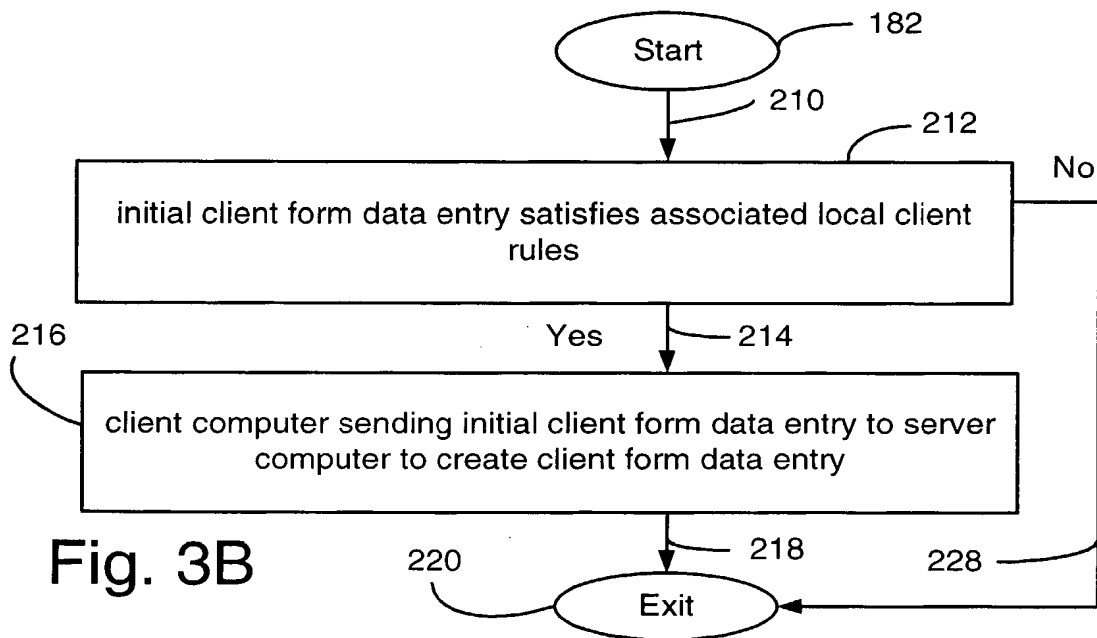


Fig. 3B

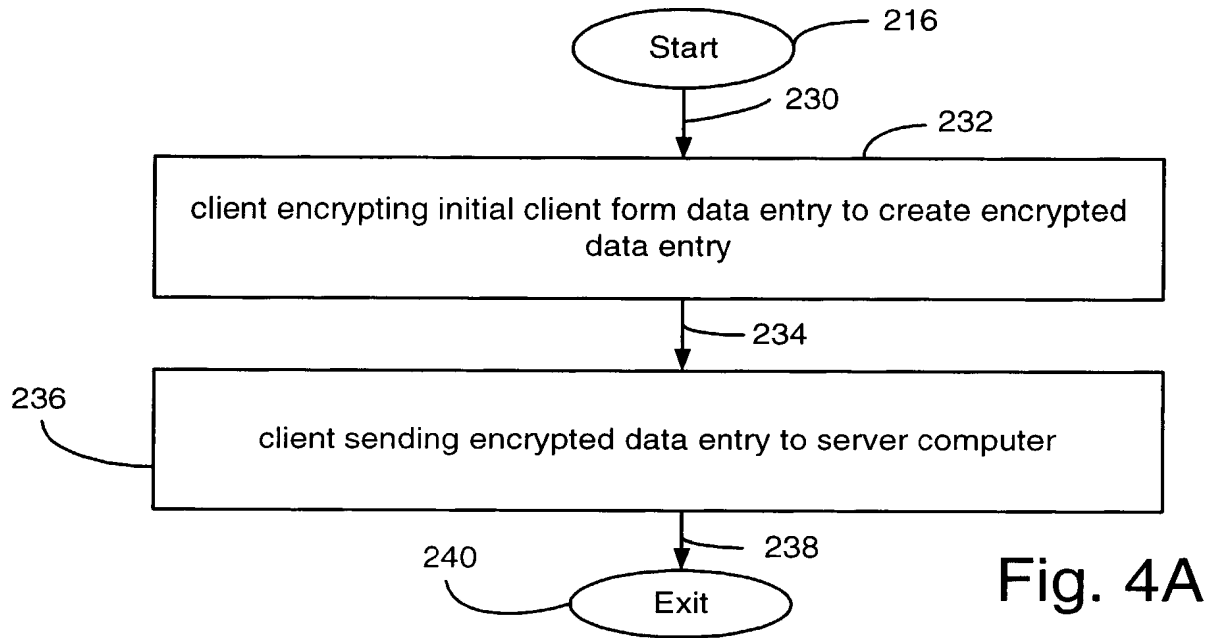


Fig. 4A

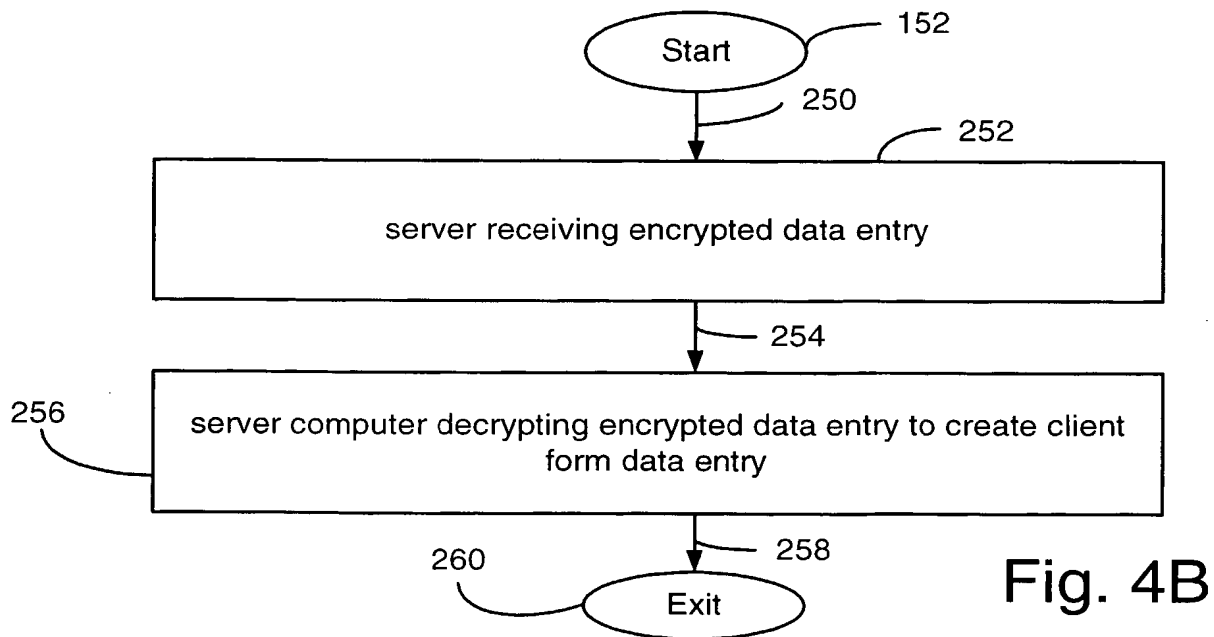


Fig. 4B

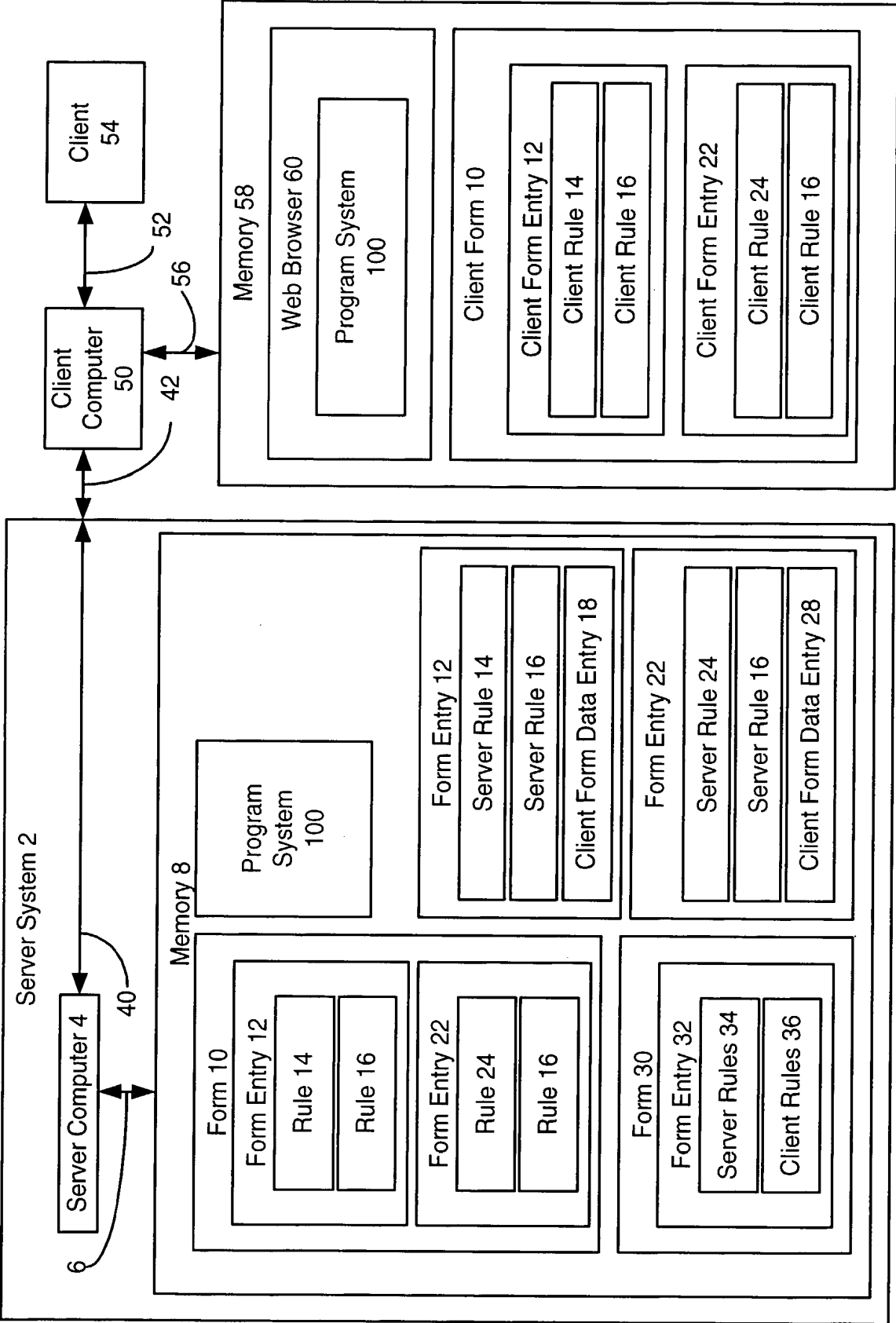


Fig. 5

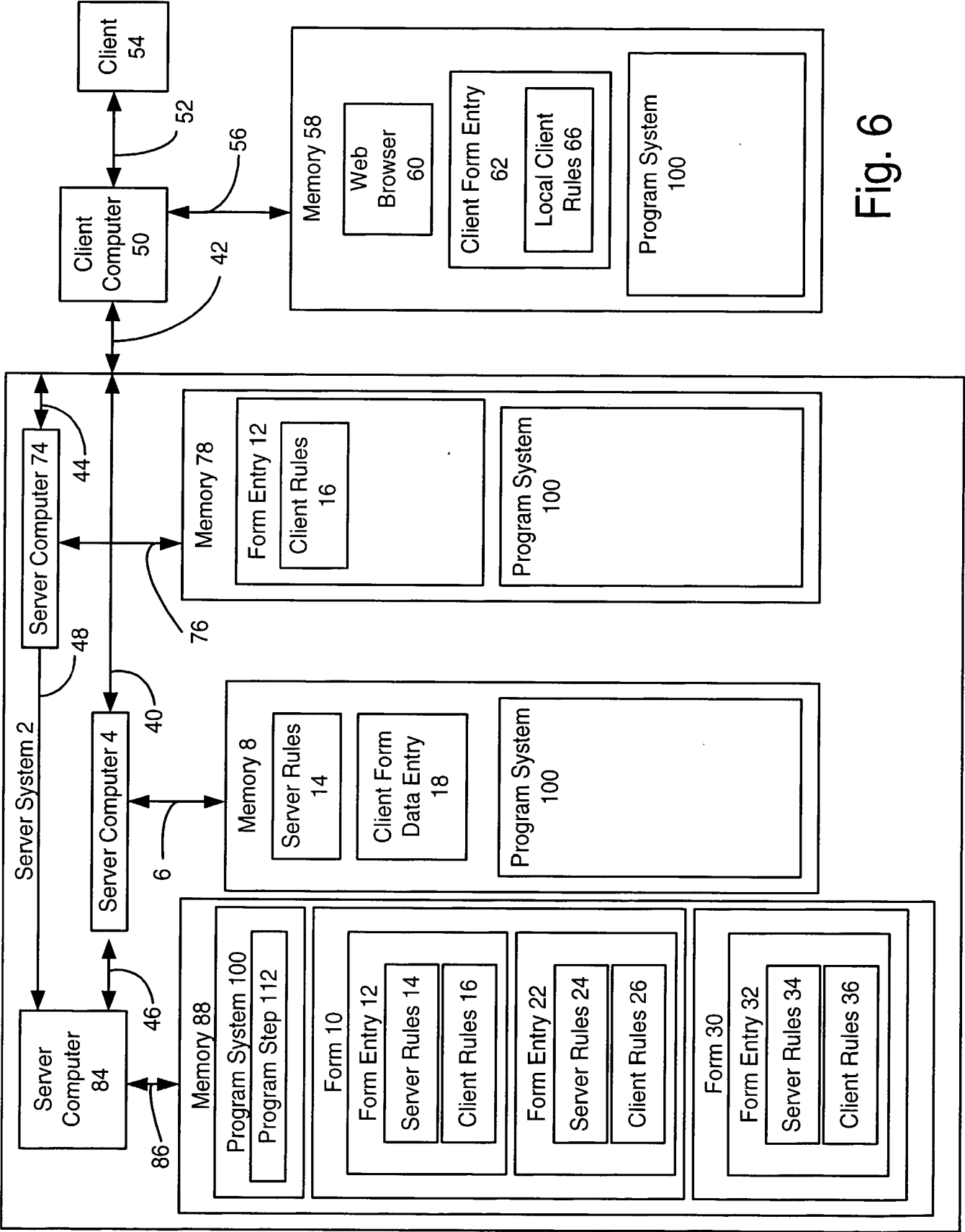


Fig. 6

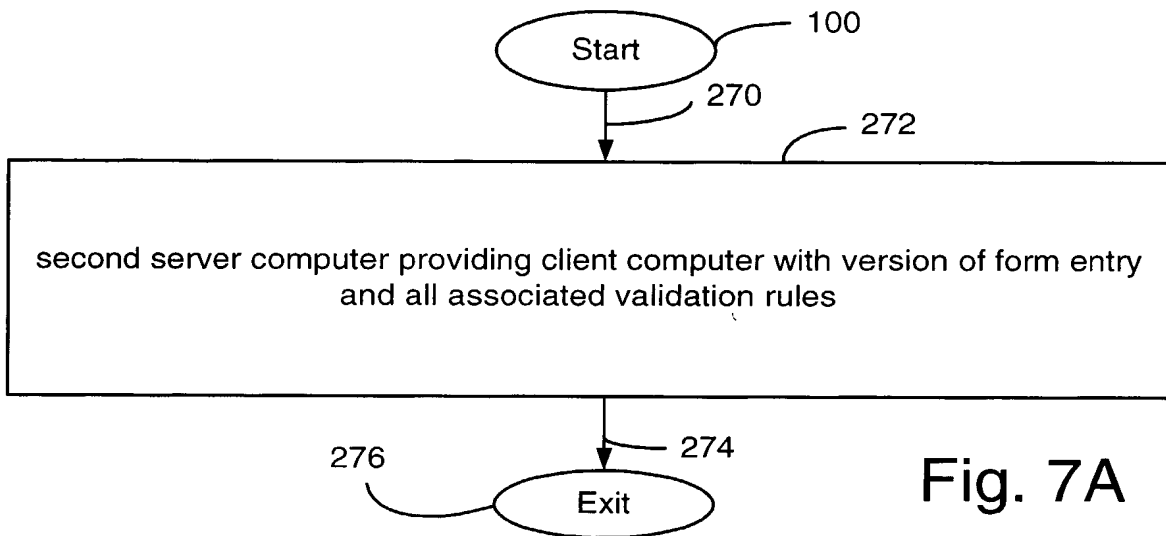


Fig. 7A

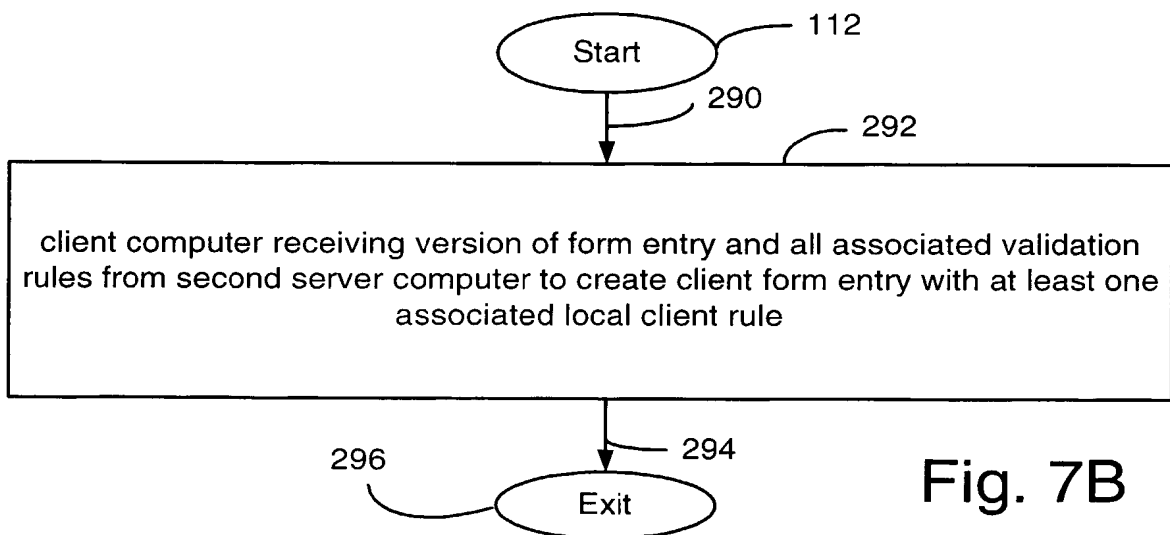


Fig. 7B

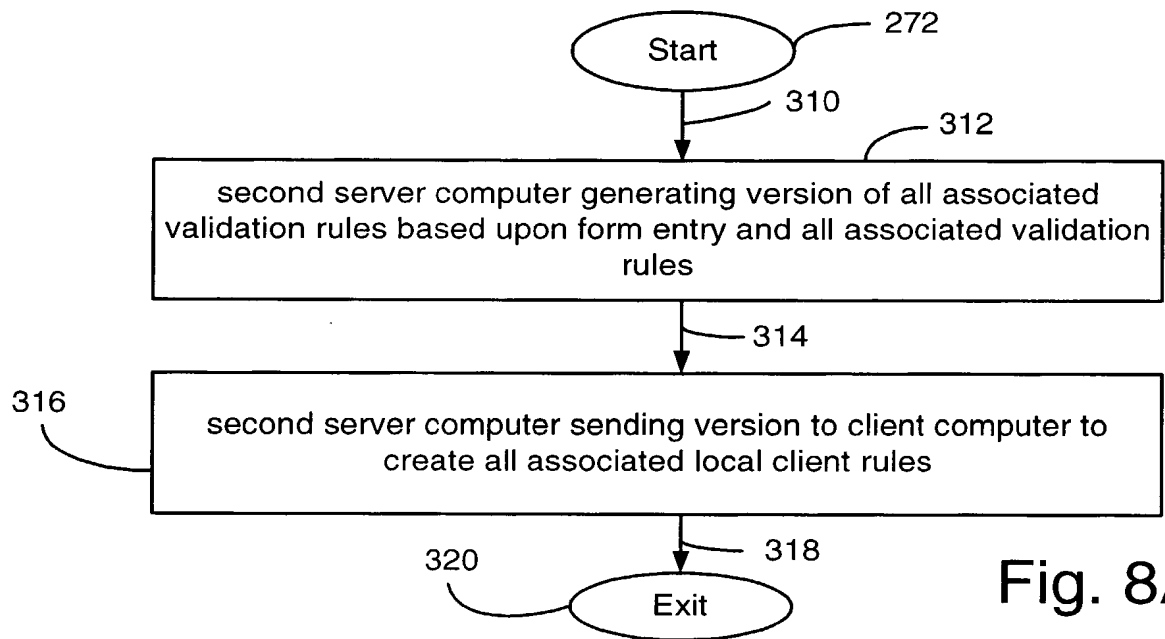


Fig. 8A

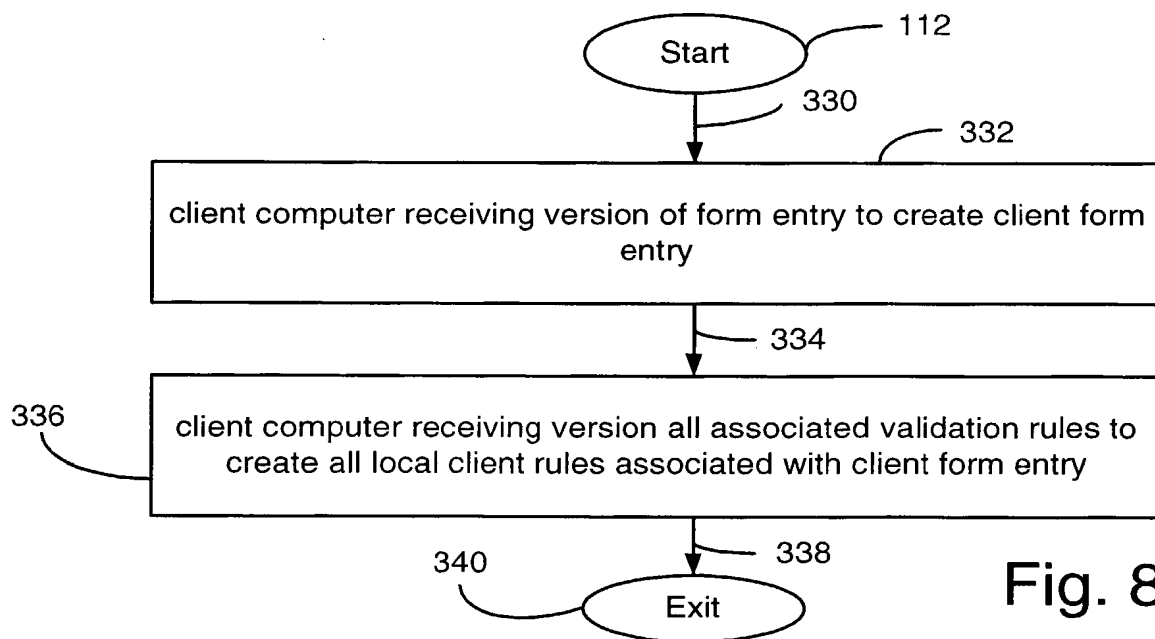


Fig. 8B

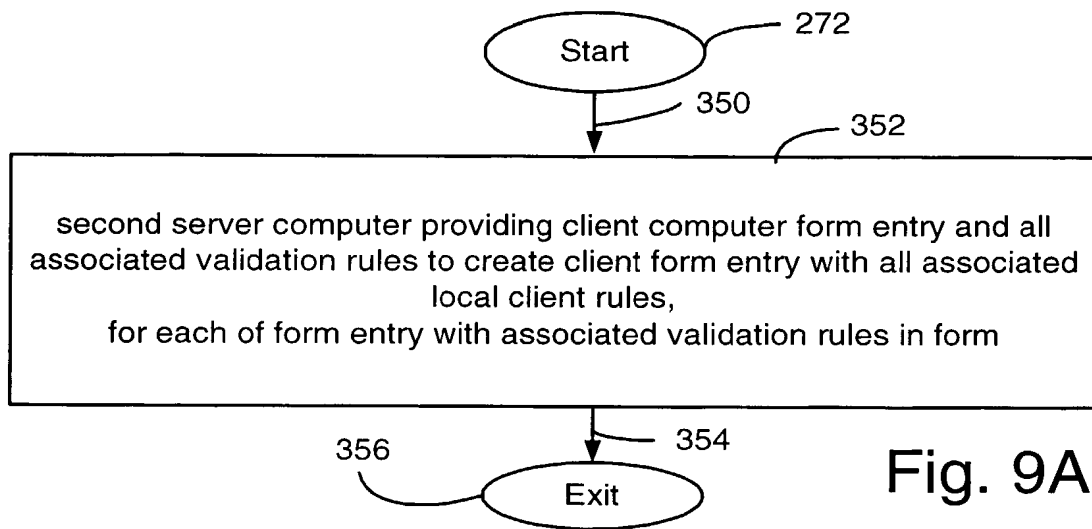


Fig. 9A

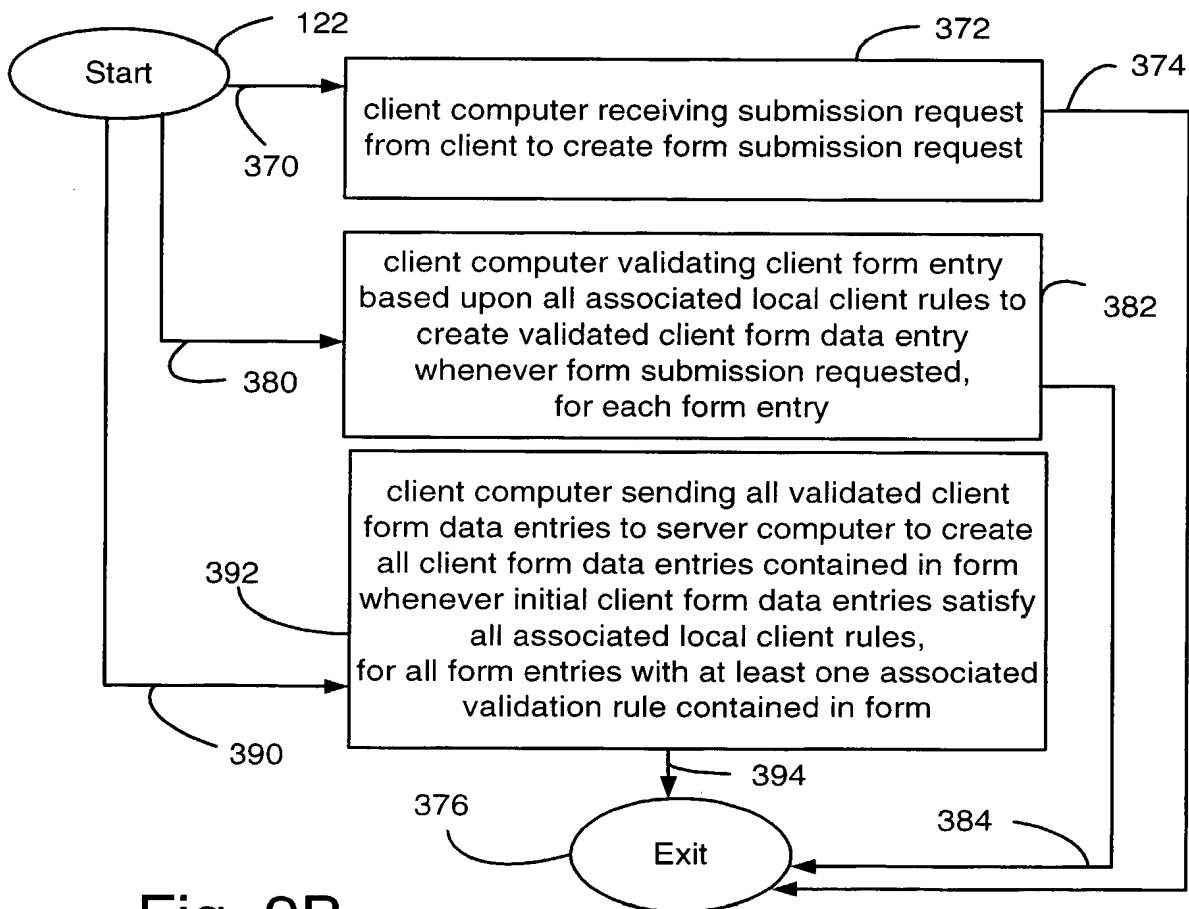
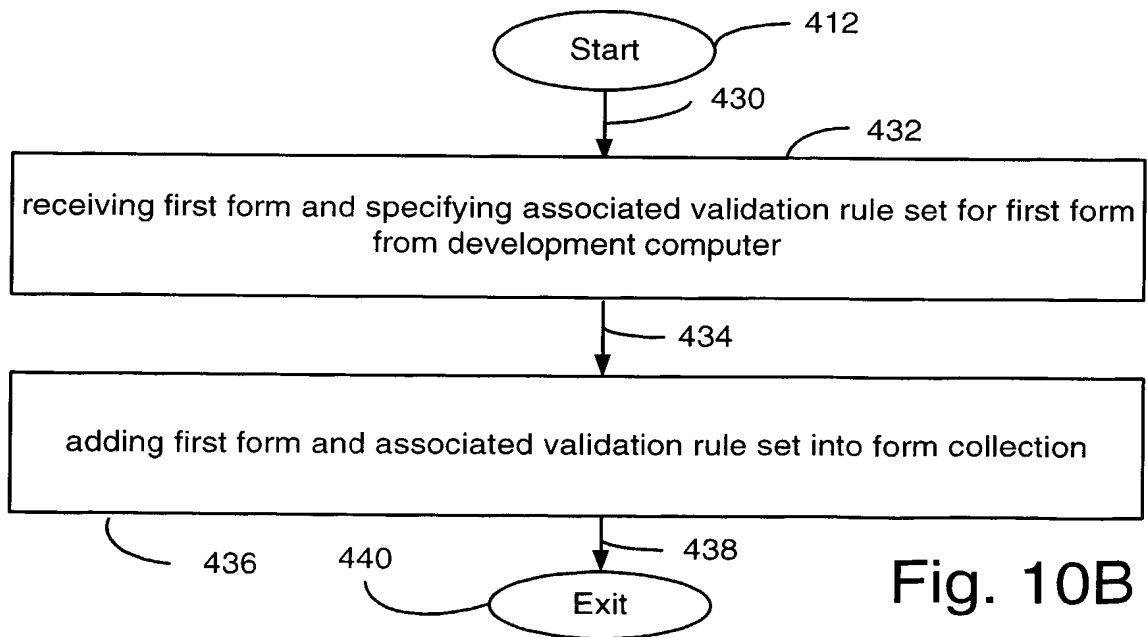
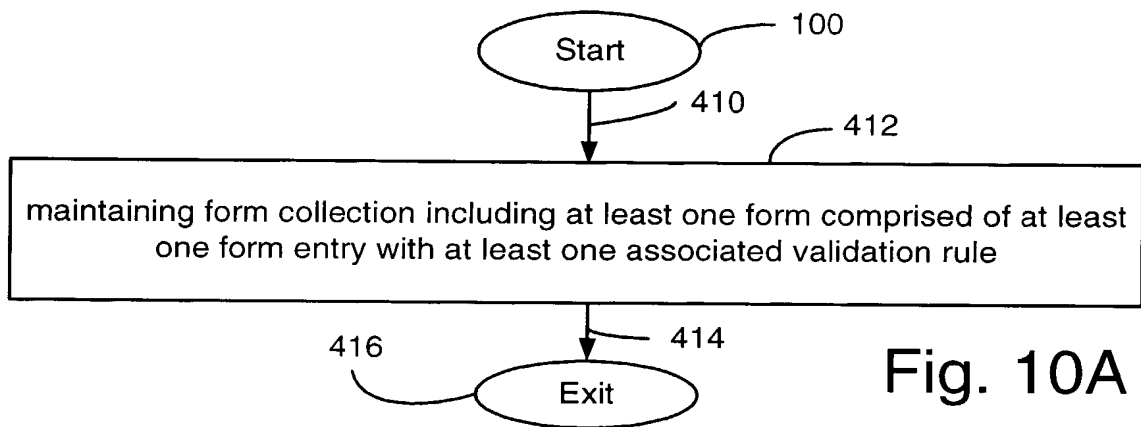


Fig. 9B





11/16

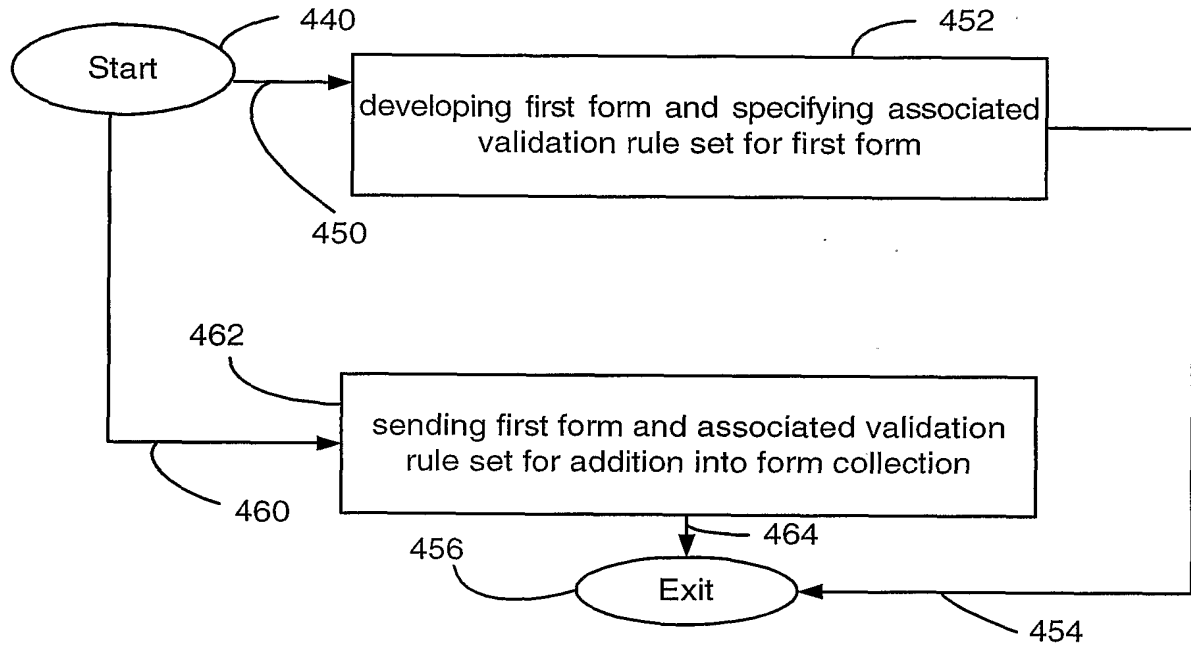


Fig. 11A

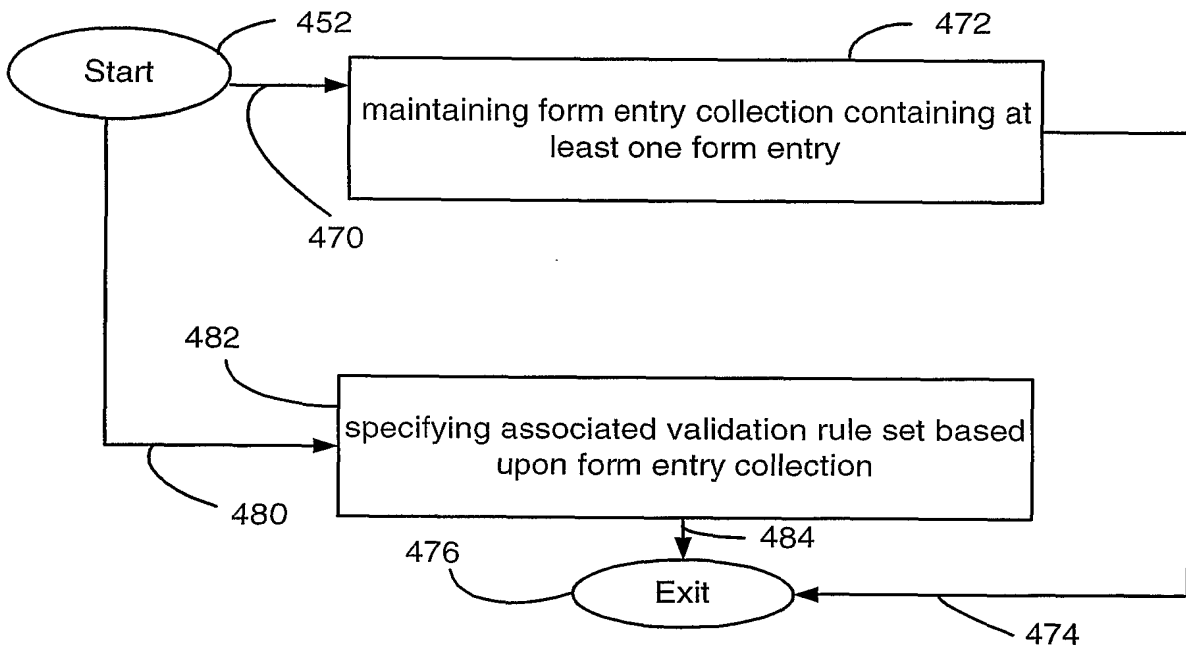


Fig. 11B

12/16

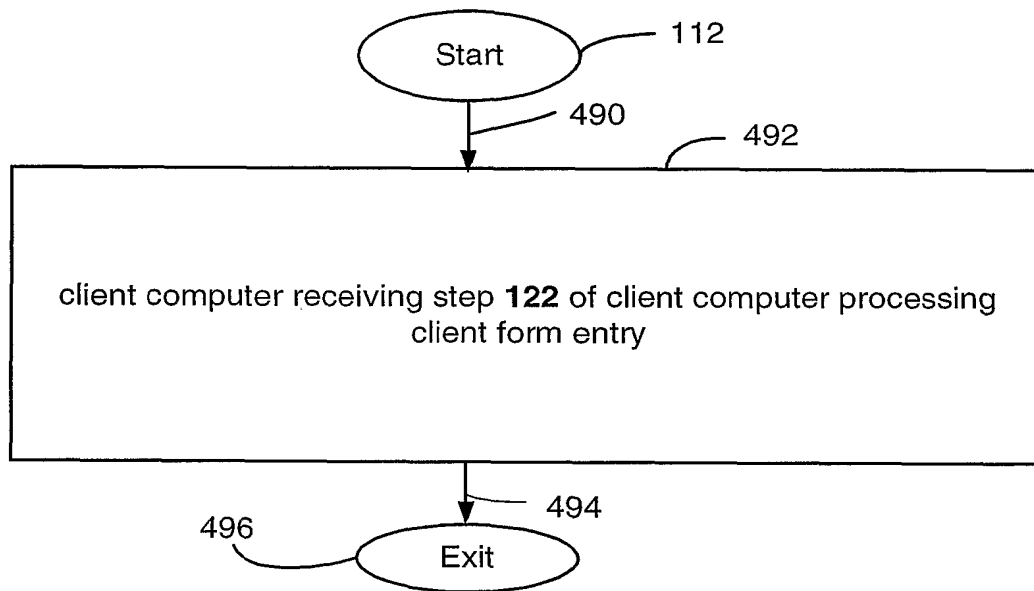


Fig. 12A

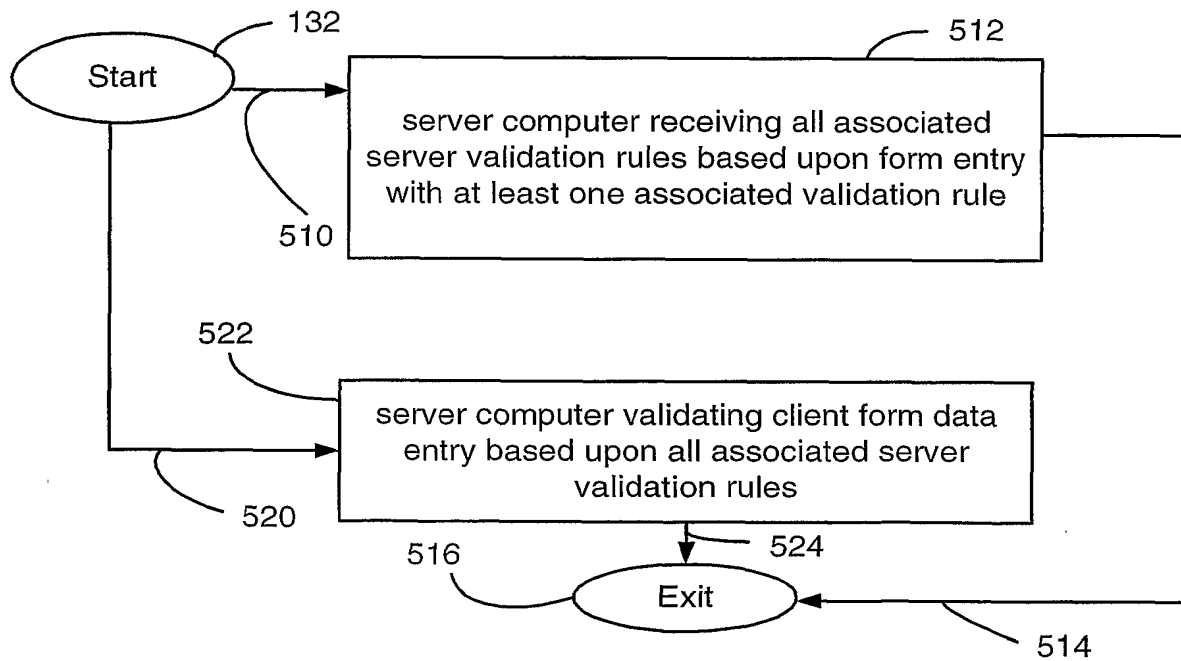


Fig. 12B

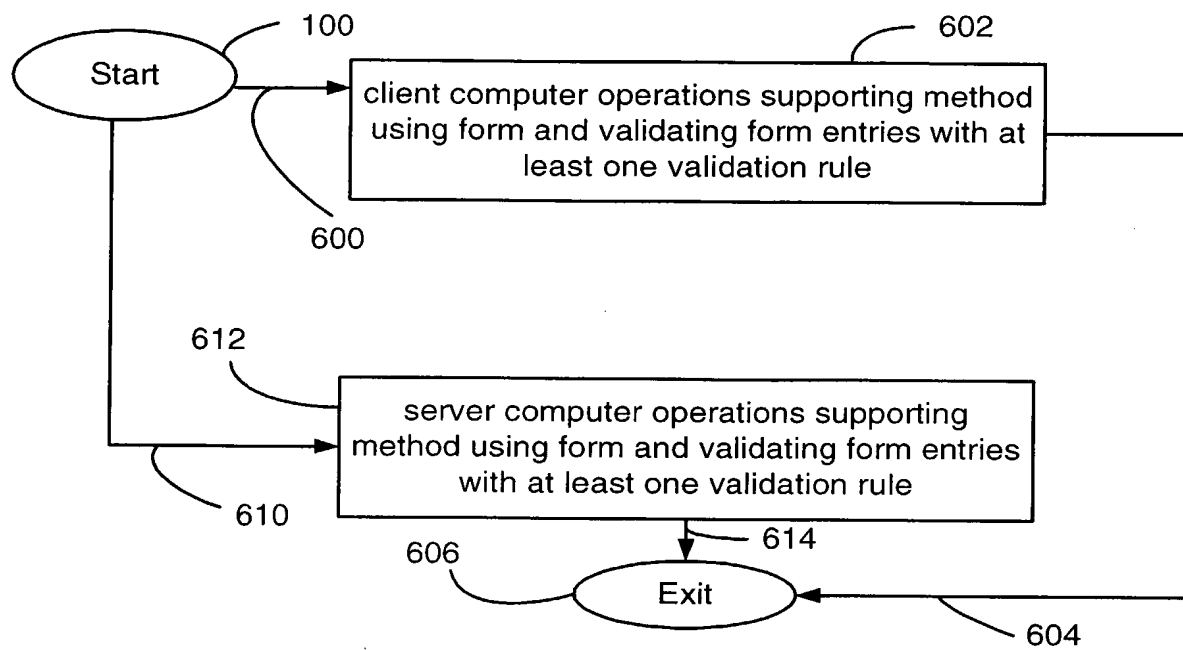


Fig. 13

14/16

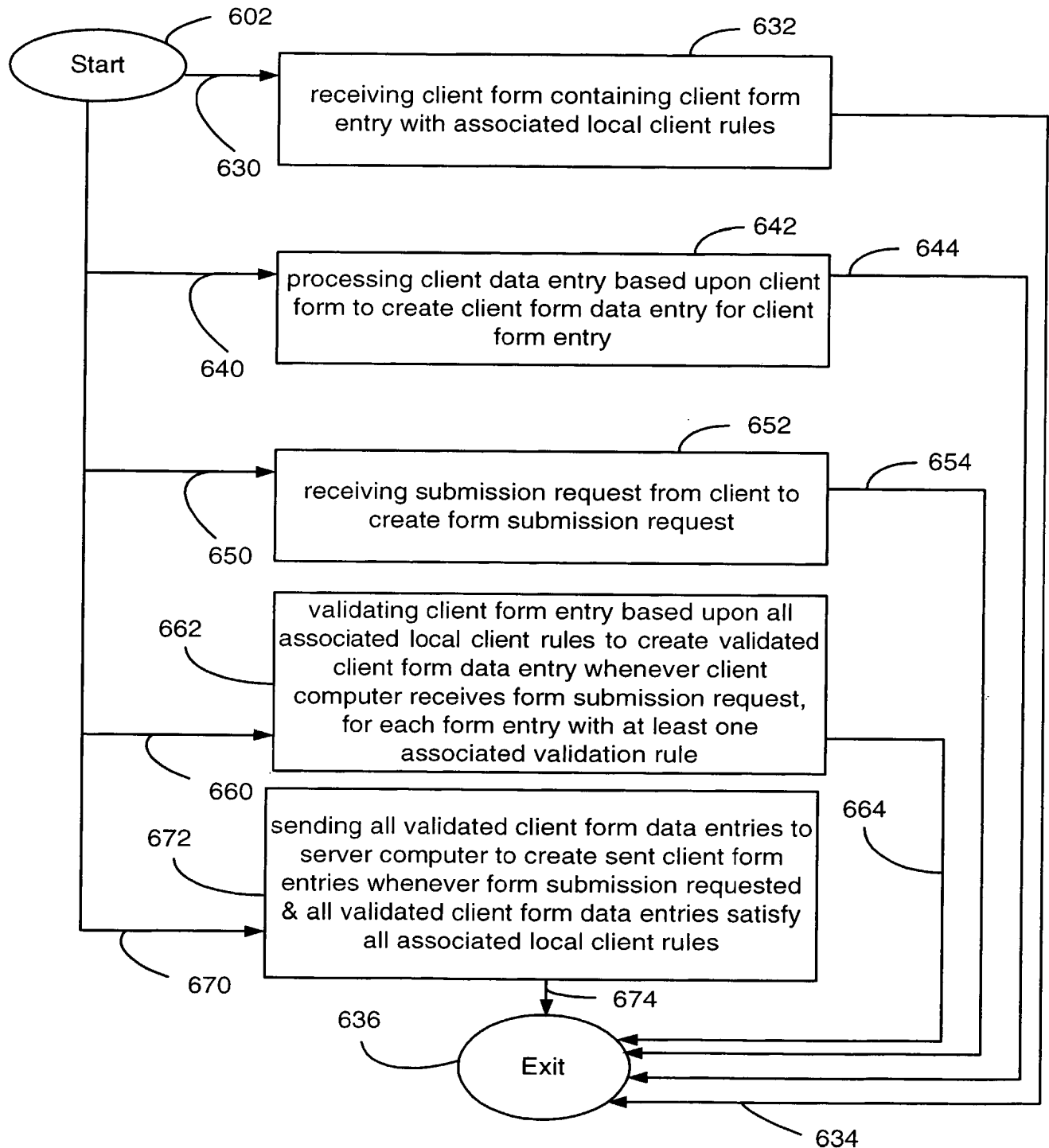


Fig. 14A

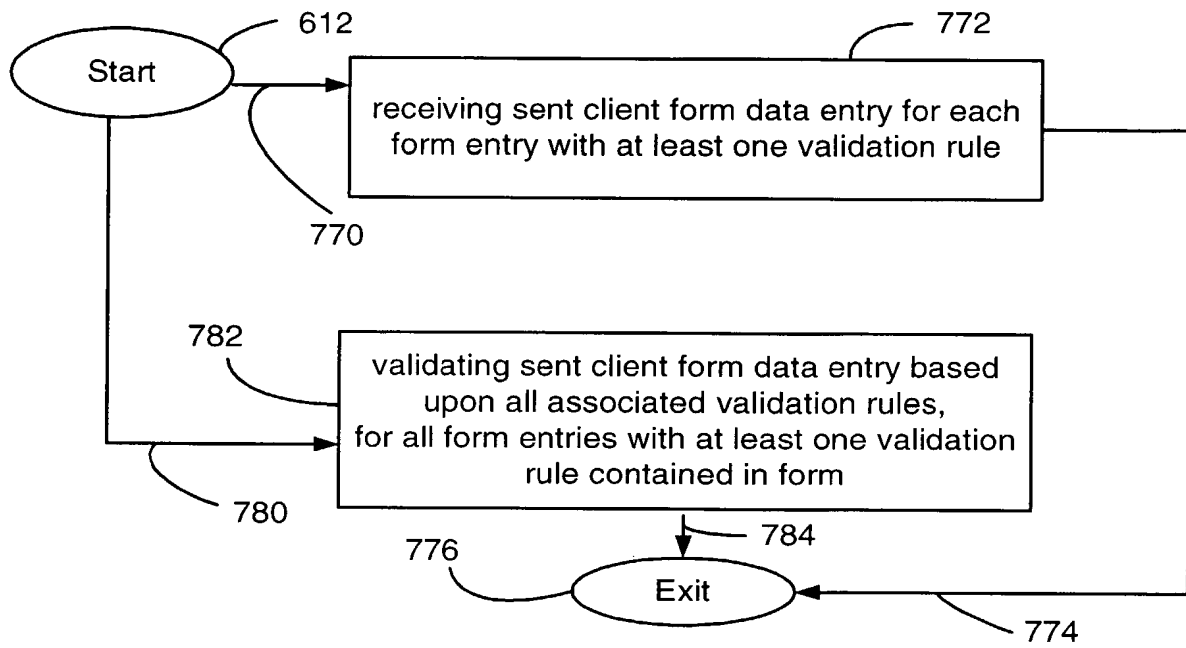


Fig. 14B

16/16

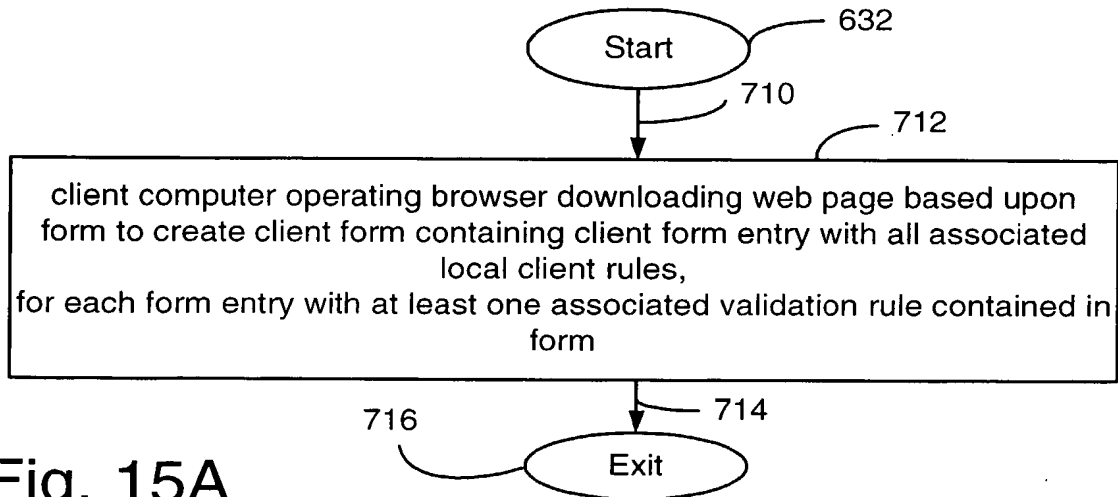


Fig. 15A

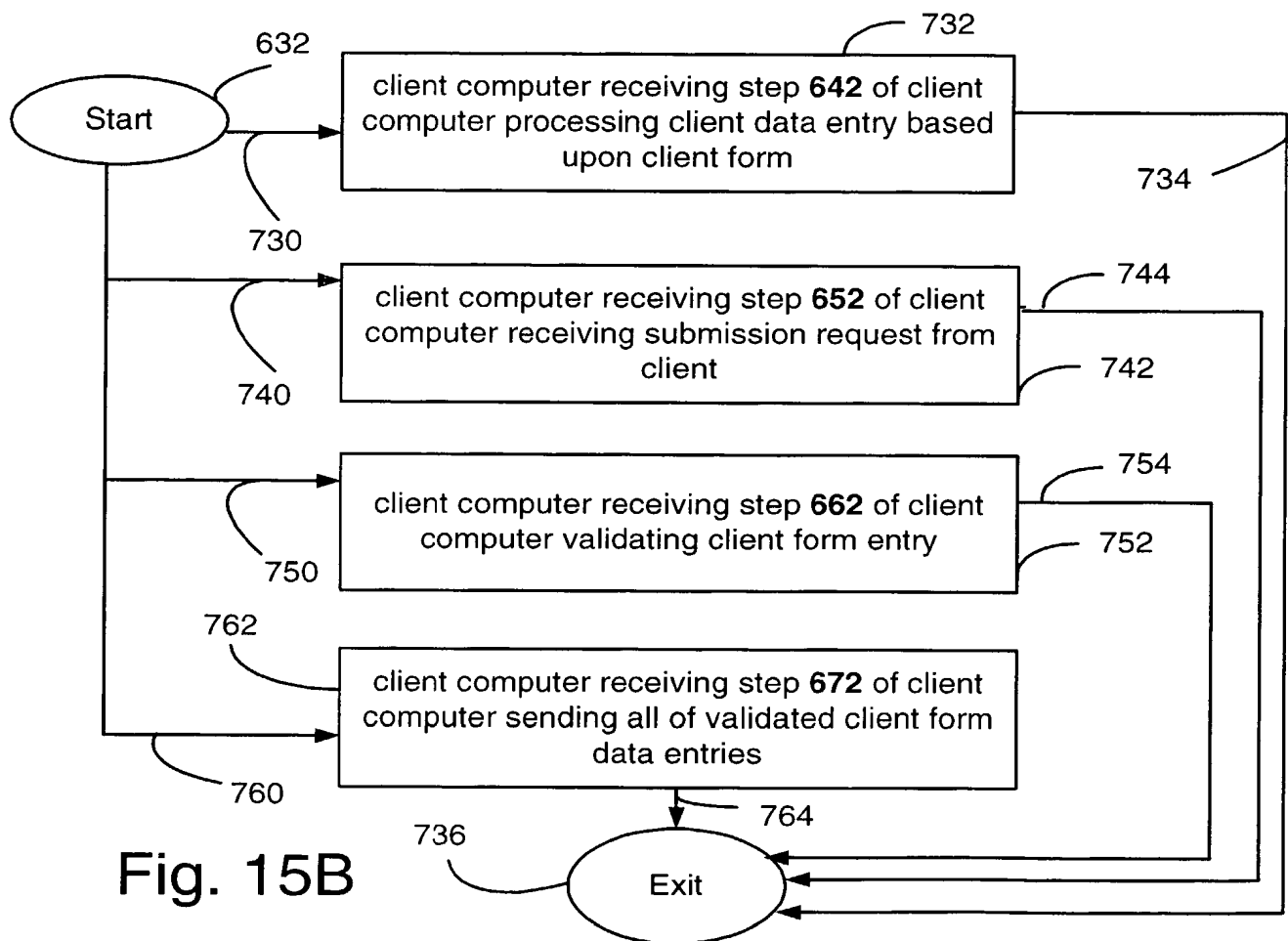


Fig. 15B